Universidade de Aveiro Departamento de Biologia
2005

Jacinto José Fonseca Pereira

Desenho de chips de DNA para o diagnóstico de infecções fúngicas

Universidade de Aveiro Departamento de Biologia
2005

Jacinto José Fonseca Pereira

Desenho de chips de DNA para o diagnóstico de infecções fúngicas

dissertação apresentada à Universidade de Aveiro para cumprimento dos requisitos necessários à obtenção do grau de Mestre em Métodos Biomoleculares Avançados, realizada sob a orientação científica do Dr. Manuel Santos, Professor Auxiliar do Departamento de Biologia da Universidade de Aveiro

o júri

presidente

Prof. Dr. António Carlos Matias Correia
professor associado da Universidade de Aveiro

Prof. Dr. José Luís Guimarães Oliveira
professor associado da Universidade de Aveiro

Prof. Dr. António Manuel Veríssimo Pires
professor auxiliar da Faculdade de Ciências e Tecnologia da Universidade de Coimbra

Prof. Dr. Manuel António da Silva Santos
professor auxiliar da Universidade de Aveiro

agradecimentos

Agradeço ao meu orientador, Prof. Manuel Santos e ao Prof. José Luís Oliveira pelo apoio científico prestado durante a preparação desta dissertação. Quero igualmente agradecer a todos os membros do grupo de Bioinformática por me terem proporcionado um bom ambiente de trabalho.

Dedico, também, um "Obrigado" especial ao Pedro, Ricardo e à Mafalda pela amizade e encorajamento.

Finalmente, uma palavra de eterna gratidão ao Senhor Mário e à Dona Teresa pelo apoio, dedicação e paciência, durante todos estes anos.

palavras-chave

Fungos Patogénicos, Diagnóstico, Microarrays, Chips, Selecção de Sondas, Bioinformática.

resumo

A tecnologia de microarrays tem uma vasta gama de aplicações em biologia molecular e medicina. Uma delas é capacidade de diagnosticar infecções patogénicas em pacientes humanos. Neste projecto, foram implementadas ferramentas bioinformáticas de selecção de sondas de DNA para desenhar chips de DNA para o diagnóstico de infecções provocadas por fungos patogénicos. O objectivo principal deste estudo é identificar, para cada espécie, um conjunto de sondas de DNA específicas. Cada sonda consiste num oligonucleótido que hibrida com apenas uma espécie fúngica descoberta até à data. O sistema foi testado utilizando 10 espécies de fungos patogénicas que apresentam elevada proximidade. Para isso, foram pesquisadas bases de dados online e extraídas as sequências de DNA pertencentes aos genes do RNA ribossomal (SSU, ITS1, ITS2, LSU), que exibem variabilidade entre espécies. Estas sequências foram de seguida submetidas a alinhamentos múltiplos e "par a par" sistemáticos, no sentido de se determinar os oligonucleótidos mais específicos para cada espécie. Este processo foi executado com recurso aos algoritmos BLAST e Clustal, a algoritmos de cálculo da temperatura de fusão e a scripts de PERL. Finalmente, as sequências foram validadas contra uma base de dados universal, para verificar a especificidade de cada sonda. Para solucionar o problema da baixa concentração de DNA fúngico em amostras clínicas, foram desenhados pares de primers para a amplificação por PCR de secções dos 4 genes de rRNA nas 10 espécies de fungos simultaneamente.

keywords

Pathogenic Fungi, Diagnosis, Microarrays, Chips, Probe Selection, Bioinformatics.

abstract

Microarray technology has a wide variety of applications in molecular biology and medicine. One of these is the diagnostics of pathogenic infections in human patients. In this project, bioinformatics tools for DNA probe selection were set up to design DNA chips for the diagnosis of infections caused by pathogenic fungi. The main objective of the study was to identify, for each fungal species, a set of specific DNA probes. Each probe consists of an oligonucleotide that hybridises with only one, presently discovered, fungal species. The system was tested using 10 closely related pathogenic fungal species. For this, online databases were searched and DNA sequence data belonging to ribosomal RNA genes (SSU, ITS1, ITS2, LSU), which exhibit variability between species, was downloaded. These sequences were then subjected to systematic pairwise and multiple alignments in order to find the most specific oligonucleotides for each species. This procedure was performed with the BLAST and Clustal algorithms, melting temperature calculation algorithms and PERL scripting. Finally, the sequences were validated against a universal database in order to verify the specificity of each probe. To overcome the problem of the low concentration of fungal DNA in clinic samples, pairs of primers were designed for the PCR amplification of sections of 4 rRNA genes in 10 species simultaneously.

Índice

Acrónimos e Abreviaturas

A – Adenina.
Å – *Angstrom*. Unidade de comprimento igual a 10^{-8} centímetros.
ASN – *Abstract Syntax Notation*.
ATP – Adenosina 5'-trifosfato.
Bioperl – Conjunto de módulos de PERL para tratamento de informação biológica.
BLAST – *Basic Local Alignment Search Tool*.
Bp – *Base pair*.
C – Citosina.
CDS – *Coding Sequence*.
CGI – *Common Gateway Interface*.
CIBEX – *Center for Information Biology Gene Expression Database*.
CPAN – *Comprehensive Perl Archive Network*.
Cy3 – *indocarbocyanine*.
Cy5 – *indodicarbocyanine*.
DDBJ – *DNA Data Bank of Japan*.
DNA – *deoxyribonucleic acid* / ácido desoxirribonucleico.
E value – *Expectation value*.
EBI – *European Bioinformatics Institute*.
EMBL - *European Molecular Biology Laboratory*.
FTP – *File Transfer Protocol*.
G – Guanina.
Gb – *Gigabase pair*.
GEO – *Gene Expression Omnibus*.
HIV – *Human immunodeficiency virus*.
HSP – *High-scoring Segment Pair*.
HTML – *HyperText Markup Language*.
ITS1 – *Internal Transcribed Spacer* 1.
ITS2 – *Internal Transcribed Spacer* 2.
IUPAC – *International Union of Pure and Applied Chemistry*.
Kb – *Kilobase pair*.
LSU – *Large Subunit* / Subunidade grande do ribossoma. Pode também ser aplicado à cadeia de rRNA 28S (ou equivalentes).
M – Unidade de Concentração molecular. Mole de Moléculas por decímetro cúbico.
MAGE – *Microarray Gene Expression*.
MAGE-ML – MAGE *Markup Language*.
MAGE-OM – MAGE *Object Model*.
Mb – *Megabase pair*.
mer – Sufixo aplicado à extensão de oligonucleótidos.

MGED – *Microarray Gene Expression Data*.
MIAME – *Minimum Information About a Microarray Experiment*.
MMDB – *Molecular Modeling Database*.
mRNA – RNA mensageiro.
MS – Microsoft ®.
N – Qualquer base; A, C, G ou T.
NCBI – *National Center for Biotechnology Information*.
nr – Base de Dados *non-redundant*.
nucl. – Nucleótidos.
oligos – Oligonucleótidos.
OMIM – *Online Mendelian Inheritance in Man*.
ORF – *Open Reading Frame*.
PCR – *Polymerase Chain Reaction*.
PERL – *Practical Extraction and Report Language*.
PIR – *Protein Information Resource*.
rDNA – Genes codificadores do rRNA. .
RID – *Request Identifier*.
RNA – *ribonucleic acid* / ácido ribonucleico.
rRNA – RNA ribossomal.
RT-PCR – *Reverse Transcriptase*-PCR.
S – Unidades de *Svedberg* de medida de centrifugação.
SGD – *Saccharomyces Genome Database*.
SIDA – Síndroma de imunodeficiência adquirida.
SNP – *Single-Nucleotide Polymorphism*.
spp. – Espécie.
SRS – *Sequence Retrieval System*.
SSU – *Small Subunit* / Subunidade pequena do ribossoma. Pode também ser aplicado à cadeia de rRNA 18S (ou equivalentes).
T – Timina.
TIGR – *The Institute for Genomic Research*.
Tm – *Melting temperature* / Temperatura de fusão.
$\mathbf{Tm_{máx\_inesp}}$ – Maior valor de Tm calculado entre todas as possíveis hibridações inespecíficas.
tRNA – RNA de transferência.
U – Uracilo.
URL-API – *Uniform Resource Locator Application Program Interface*.
XML – *Extensible Markup Language*.
ΔG – Variação da Energia Livre.
ΔH – Variação da Entalpia.
ΔS – Variação da Entropia.

Capítulo I: Introdução

O objectivo desta tese é descrever os procedimentos conducentes ao desenvolvimento de uma nova metodologia de diagnóstico molecular, baseado na tecnologia de *chips* de DNA, capaz de identificar as espécies de fungos patogénicos responsáveis por infecções em pacientes humanos.

Esta tese está estruturada em cinco capítulos: Introdução, Metodologias, Resultados, Discussão e Conclusões e Trabalho Futuro. Na Introdução, é descrita a biologia dos fungos patogénicos, a sua importância clínica e as metodologias tradicionais de diagnóstico das suas infecções em humanos. De seguida, são explicadas as bases do diagnóstico molecular e referidos os fundamentos teóricos da tecnologia de *chips* de DNA. Por último, é referida a importância das aplicações bioinformáticas na genómica e no diagnóstico molecular. No capítulo das Metodologias, são descritos exaustivamente todos os procedimentos e sistemas bioinformáticos de desenho de *chips* de DNA implementados nos estudos decorrentes desta tese. No terceiro capítulo, são apresentados os resultados da aplicação dos referidos sistemas, sob a forma de conjuntos validados e caracterizados de sequências específicas de DNA. Nos dois últimos capítulos, são discutidos os principais factores considerados no decorrer da elaboração desta tese, são expostas as conclusões finais e são indicadas algumas vias complementares ou alternativas de prosseguimento de trabalhos.

A 1. Biologia dos fungos patogénicos e metodologias de identificação.

Os fungos são organismos eucariótas, heterotróficos e essencialmente aeróbios com limitadas capacidades anaeróbias. São conhecidas cerca de 700000 espécies [1] mas segundo algumas estimativas o seu número total será superior a 1.6 milhões [2]. Estes seres apresentam uma grande variedade de estilos de vida, podendo ser multi- ou unicelulares. Possuem paredes celulares com quitina e membranas plasmáticas com esteróis. Outras características, comuns a todos os eucariótas, são: uma membrana a envolver um núcleo (com vários cromossomas), vários organelos membranares incluindo mitocôndrias e vacúolos, regiões não codificantes do DNA chamadas intrões e ribossomas do tipo 80S em contraste com os 70S dos seres procariotas (ver secção 2 para mais detalhes) [1-3].

Os fungos são seres heterotróficos porque necessitam de compostos orgânicos pré-formados como fonte de energia e fonte de carbono para a síntese celular. Obtêm os nutrientes essencialmente por absorção de materiais orgânicos e inorgânicos solúveis [4]. A digestão de alimentos é realizada exteriormente por enzimas hidrolíticas libertadas nas proximidades do organismo.

A reprodução destes organismos faz-se por via sexuada ou assexuada, com a formação de esporos. Estes podem ser dos mais variados tipos e formas, de acordo com as suas funções. A Reprodução sexuada ocorre com a fusão de dois núcleos haplóides (Cariogamia), seguida da divisão meiótica do núcleo diplóide. A reprodução assexuada resulta da divisão de um núcleo por mitose [1].

Os fungos podem apresentar uma estrutura filamentosa, com os longos filamentos, que crescem por extensão apical, a serem designados por hifas. As hifas sofrem várias ramificações formando uma rede – o micélio. Para além desta formação micelar, os fungos podem também desenvolver-se sob a forma de leveduras - espécies unicelulares que produzem células filhas por *budding* (a nova célula forma-se a partir de uma protuberância da célula-mãe) ou por *binary fission* (formação de duas células novas pela divisão completa de uma) [5]. Os fungos adoptam uma das duas formas de crescimento de acordo com as condições ambientais do seu habitat. Determinados fungos podem mesmo alternar entre uma "fase micelar" e uma "fase de levedura" em resposta a condições ambientais [3].

A taxonomia dos fungos está em constante processo de actualização. No passado, a classificação dos seus maiores grupos e os relacionamentos entre eles eram baseadas na

comparação das morfologias e padrões de desenvolvimento das suas estruturas reprodutivas. Actualmente, estes estudos baseiam-se em análises de sequências de ácidos nucleicos, com particular ênfase na sequência de DNA que codifica o RNA ribossomal [3].

O fungos (numa visão mais tradicional) podem ser divididos em: *Eumycota* e *Myxomycota*, correspondendo a fungos verdadeiros e myxomycetes, respectivamente. Os fungos verdadeiros são aquelas espécies que possuem hifas (ou estão nitidamente relacionadas com espécies que as possuem), possuem parede celular durante grande parte do seu ciclo celular e obtêm os nutrientes unicamente por absorção. Os myxomycetes não formam hifas não possuem parede celular durante a fase de crescimento e são capazes de ingerir alimento por um fenómeno de fagocitose [2]. Deste modo, os myxomycetes não são considerados como verdadeiros fungos, estando, em alguns aspectos mais próximos do Reino Protista [2], estando actualmente fora do Reino Fungi.

Os fungos, anteriormente referidos como pertencentes à divisão *Eumycota*, são divididos, segundo regras actuais, nos seguintes grupos principais: *Chytridiomycota*, *Zygomycota*, *Ascomycota*, *Deuteromycota*, *Basidiomycota* e *Oomycota* [3].

As leveduras são um caso particular na classificação dos fungos pois não são formalmente uma unidade taxonómica, mas uma forma de desenvolvimento (em todo ou em parte do seu ciclo de vida, ou apenas em condições ambientais particulares) adoptada por uma gama de espécies sem parentesco. Neste grupo são incluídos vários (o seu número varia conforme a autoridade e a definição utilizada) géneros de fungos [5]. No entanto, continua a ser útil referir leveduras como termo de trabalho, porque têm muito em comum, umas com as outras, em relação à estrutura, habitat, importância prática e métodos de identificação [2].

O estudo científico dos fungos é designada por micologia [5]. No seu início, foi considerada um ramo da botânica, devido à tradicional integração dos fungos no reino das plantas [2]. Actualmente, por si só, é uma área de investigação profundamente desenvolvida dada a importância dos fungos, nomeadamente na medicina, agricultura e indústria [2]. As suas funções benéficas para o homem incluem, entre muitas outras, o seu uso na alimentação (colheita e cultivo de cogumelos, fermentação de bebidas alcoólica e fermento do pão), na preparação industrial de antibióticos e na produção de vacinas e hormonas (por técnicas de recombinação de DNA e clonagem). Por outro lado, os fungos

são os responsáveis por algumas das mais importantes doenças em plantas, promovem a deterioração de alimentos armazenados e provocam infecções patológicas em humanos [2].

1.1. Fungos patogénicos.

De todas as espécies actualmente conhecidas, aproximadamente 300 são dadas como responsáveis por infecções (micoses) em humanos. Embora seja um número relativamente pequeno (comparado com a incidência de doenças de origem micológica em plantas), as consequências destas patologias podem ser muito graves, principalmente em indivíduos imunodeprimidos – como é o caso dos doentes com SIDA ou dos pacientes sujeitos a tratamentos com administração de medicamentos imunosupressores [3].

A infecção pode ser definida como a entrada de fungos em tecidos do hospedeiro seguida da sua multiplicação. A infecção pode não ser detectada clinicamente, ou então pode resultar numa doença provocada por lesão celular (metabolismo competitivo), elaboração de metabolitos tóxicos, replicação do fungo ou resposta imunitária (mediada por células, por anticorpos ou ambos) [1].

A origem e o processo de infecção de seres humanos (e outros animais de "sangue quente") por fungos patogénicos são diversificados. Os agentes que provocam micoses superficiais, como "pé de atleta" e a tinha são espécies comuns, em que o parasitismo é o seu modo natural de crescimento. Este fungos, denominados dermatófitas, englobam várias espécies dos género *Trichophyton*, *Microsporum*, e *Epidermophyton*. São filamentosos, capazes de digerir e obter nutrientes da queratina – proteína essencial da pele, unhas e cabelo – razão pela qual têm necessidade de colonizar humanos e outros animais [6]. Normalmente, os dermatófitas penetram nas camadas superficiais da pele através de pequenos ferimentos e desenvolvem infecções de gravidade variada. No entanto, não são capazes de infectar mais profundamente o hospedeiro [1].

Outros fungos apresentam-se como comensais inofensivos das mucosas do hospedeiro, e em certas condições tornam-se invasores patogénicos – um exemplo clássico é a levedura *Candida albicans* [3]. Outros, ainda, são considerados patogénicos oportunistas, pois são normalmente encontrados como saprófitas no solo, em plantas ou em resíduos animais, mas podem igualmente crescer nos pulmões (após inalação de esporos) e invadir profundamente os tecidos de hospedeiros gravemente doentes [3]. Alguns exemplos são as espécies *Aspergillus fumigatus* e *Cryptococcus neoformans*.

A classificação das infecções por fungos patogénicos pode também ser feita de acordo com a via de aquisição: exógena ou endógena [1]. Na primeira, o organismo infeccioso é transmitido pelas vias aéreas, cutâneas ou percutâneas. A segunda é originada pela colonização ou reactivação de um fungo de uma infecção latente.

A identificação da espécie de fungo causadora de uma infecção é muitas vezes realizada por observação directa ao microscópio de amostras de tecidos do paciente ou de culturas realizadas a partir destas (ver secção 1.2). Da observação da morfologia dos fungos patogénicos em culturas, resulta outra das classificações destas espécies: 1) leveduras e organismos semelhantes monomórficos; 2) fungos termicamente dimórficos; 3) bolor termicamente monomórfico. O primeiro grupo apresenta estrutura de leveduras tanto em culturas a 25-30 ºC como a 35-37 ºC (se ocorrer crescimento a esta temperatura). O segundo grupo exibe estrutura filamentosa quando cultivado a 25-30 ºC e estrutura do tipo "levedura" a 35-37 ºC. As espécies do terceiro grupo são filamentosas quando as culturas se fazem a qualquer uma destas temperaturas.

O estado imunológico do hospedeiro humano é um dos principais factores que determinam se um fungo irá provocar uma infecção e a gravidade da mesma. Como já foi referido, os indivíduos imunodeprimidos são os principais afectados por estas patologias. Nos indivíduos saudáveis, a resposta imunitária do organismo a uma invasão fúngica é mediada quer por células quer por anticorpos. No entanto, pensa-se que a primeira é de maior importância, porque foi observado que indivíduos com deficiências na resposta mediada por células sofrem infecções mais graves do que indivíduos com produção deficiente de anticorpos [1]. As barreiras não específicas primárias são, no entanto, as primeiras defesas do organismo humano contra a invasão de fungos patogénicos e incluem a pele intacta, as membranas mucosas (estão cobertas de fluidos que contêm substâncias antifúngicas e possuem cílios nas suas células epiteliais que removem os microorganismos) e a competição com a flora bacteriana normal [1].

Dentro do grupo das leveduras, as duas principais espécies causadoras de invasões patológicas no homem são a *Candida albicans* e *Cryptococcus neoformans* [7].

1.1.1. Tipos de infecção.

Se o fungo ultrapassar as defesas do organismo humano provocará uma doença proporcional à sua virulência e à vulnerabilidade do hospedeiro. A diferentes espécies

correspondem diferentes processos de infecção com graus de gravidade variáveis. De seguida são referidos algumas doenças e espécies de fungos patogénicos que as originam.

1.1.1.1. Candidíase.

A Candidíase é a doença provocada pela infecção de um indivíduo com uma das espécies do género *Candida*. A infecção com *C. albicans* é a micose mais comum detectada em seres humanos, sendo uma preocupação constante em estabelecimentos clínicos, no processo de diagnóstico de pacientes e na detecção de contaminações hospitalares.

A espécie *C. albicans* é diplóide, sem fase sexual conhecida e pode apresentar dimorfismo durante o seu ciclo de vida. Quando na flora normal, existe sob a forma de levedura, mas torna-se filamentosa (um pseudomicélio que gera novas células de leveduras por *budding*) assim que invade o hospedeiro [1].

É um comensal muito comum dos humanos, encontrando-se nas membranas mucosas da boca, intestino e vagina. Na grande maioria dos casos, a sua presença é benigna, por acção das defesas naturais do hospedeiro e pela competição que estabelece com determinadas bactérias. Apenas quando algum factor interrompe este equilíbrio é que *C. albicans* provoca infecções prejudiciais à saúde. Nestes casos, invade as mucosas causando irritação local e em casos extremos pode mesmo crescer no corpo de forma sistémica, com consequências fatais [3].

Os factores que podem promover a infecção por este fungo são variados e estão, obviamente, relacionados com a condição imunológica do indíviduo afectado. As grávidas ou as mulheres sob efeito de contraceptivos orais, devido a uma maior predisposição hormonal, são mais susceptíveis de sofrer vulvo-vaginites por infecção com *C. albicans* – altos níveis de progesterona promovem a adesão deste fungo às células epiteliais. A infecção de recém-nascidos (ainda não desenvolveram uma população microbiana equilibrada na boca) com este fungo, originando aftas na boca e garganta, tem frequentemente origem num parto realizado através de um canal vaginal infectado. As infecções intestinais costumam ocorrer em indivíduos sujeitos a terapia antibacteriana (destrói a flora bacteriana e *C. albicans* prolifera). Foi, também, observado que pessoas com elevados níveis de stress estão mais sujeitas a sofrer infecções por *C. albicans*. Todas estas infecções, superficiais e relativamente moderadas, são invasões localizadas das

mucosas que ocorrem quando a *C. albicans* sofre metamorfose de levedura para a forma filamentosa [3]. Em situações extremas (doentes em fases avançadas de cancro, SIDA ou diabetes), *C. albicans* pode tornar-se invasiva – através de catéteres contaminados – e originar infecções profundas e sistémicas nos tecidos do hospedeiro. Os principais alvos destes casos gravíssimos de candidiasis incluem os rins, fígado, baço, cérebro, olhos e coração [1].

A *C. albicans*, em condições normais, tem um comportamento de comensal benigno mas possui determinados factores de virulência que, em condições propícias, lhe permitem iniciar um comportamento patológico. Estes incluem a sua elevada capacidade de adesão a células epiteliais e a possibilidade de alterar a sua forma de leveduriforme para micelar. Deste modo, *C. albicans* adere às células das mucosas do hospedeiro, iniciando, de seguida, a formação de hifas que lhe permitirão invadir essas mesmas células. A capacidade de excretar hidrolases é também considerada um factor de virulência. Estas enzimas quebram as barreiras ao seu crescimento e inactivam as defesas do hospedeiro. Outra característica que beneficia a sua patogenicidade em humanos é a capacidade de infectar tecidos em ambientes fisiológicos diversos; por exemplo, o pH do sangue é neutro enquanto que o da vagina é ácido [8].

Outras espécies de *Candida* são igualmente patogénicas, sendo de destacar a *C. glabrata*, *C. Krusei*, *C. parapsilosis* e *C. tropicalis*. Tal como a *C. albicans*, com a qual são regularmente detectadas, estas espécies podem provocar uma grande variedade de infecções, desde doenças superficiais até micoses disseminadas e mortais [8].

A *C. glabrata* infecta vários orgãos, especialmente o tracto urinário, mucosas e pulmões de doentes com diabetes ou tumores avançados, indivíduos malnutridos e nascituros. Os factores que favorecem a infecção são a aplicação de cânulas, catéteres intravasculares, cirurgia vascular, mecanismos de ventilação e perfuração gástrica [9].

A candidiasis provocada por *C. Krusei* tem particular incidência no sangue de doentes com SIDA, leucémia, linfomas, ou em indivíduos que receberam transplante de medula óssea. A infecção é promovida pela *neutropenia*, imunosupressão, ou administração profiláctica de fluconazole [9].

A levedura *C. parapsilosis* infecta principalmente o sangue, o peritoneu, os tecidos em contacto com catéteres intravenosos de doentes imunodeficientes, com insuficiências renais avançadas ou de bebés prematuros [9].

A *C. tropicalis*, tal como a *C. albicans*, pode infectar uma grande variedade de tecidos, atingindo principalmente doentes com deficiências a nível das defesas imunitárias. Pode também afectar indivíduos sem qualquer evidência de doença [6].

1.1.1.2. Aspergilose.

A Aspergilose refere-se à micose provocada pela infecção de um organismo hospedeiro com fungos do género *Aspergillus* especialmente as espécies *A. fumigatus*, *A. flavus* e *A. niger*. Estes fungos, normalmente saprófitas, produzem conídia abundantemente (esporos assexuados), que são estruturas transportadas pelo ar e que permitem a sua infiltração no hospedeiro através das vias respiratórias, alojando-se nos pulmões. A partir daqui, em condições propícias, podem provocar uma infecção disseminada e atingir orgãos vitais como o cérebro, rins, fígado, coração e ossos. A via de entrada no organismo humano pode também ser através de lesões na pele. Mais uma vez, os indivíduos com sistemas imunitários debilitados (como por exemplo os pacientes com neutropenia resultante de quimioterapia) são os mais susceptíveis de sofrer Aspergilose, dado que em indivíduos saudáveis é muito pouco frequente a ocorrência deste tipo de infecções [1].

1.1.1.3. Cryptococose.

A Cryptococose é outro tipo de micose provocada por fungos oportunistas, neste caso a levedura haplóide *Cryptococcus neoformans*. É uma das doenças mais mortíferas em doentes com SIDA e frequentemente provoca pneumonia e/ou meningites [1].

C. neoformans possui um anamorfo que representa a sua fase sexual – *Filobasidiella neoformans*. Encontra-se normalmente como saprófita em dejectos de aves e produz esporos que se pensa serem a fonte de infecção de humanos - através das vias respiratórias [3]. Este fungo pode crescer nos pulmões e, caso a infecção perdure, dissemina-se pelo sistema nervoso central, desenvolvendo-se no córtex cerebral, tronco cerebral, cerebelo e meninges, com consequências fatais.

Um dos seus factores de virulência é a espessa cápsula que envolve toda a célula da levedura e que evita a sua ingestão pelos leucócitos. Outro determinante patogénico é a

excreção de uma enzima que oxida compostos fenólicos (fenoloxidase), que produz melanina que é depositada nas paredes da levedura protegendo-a de agentes oxidantes [3].

1.1.1.4. Outras micoses importantes.

Outras micoses oportunistas são a Zygomicose (provocada pelas espécies dos géneros *Rhizopus*, *Rhizomucor*, *Absidia* e *Mucor*), a Phaeohyphomicose (causada por vários fungos de pigmentação escura, pouco frequente mas com consequências potencialmente fatais) [1].

As espécies *Coccidioides immitis*, *Histoplasma capsulatum*, *Paracoccidioides brasiliensis*, *Blastomyces dermatitidis* e *Pneumocystis carinii* têm um comportamento infeccioso semelhante a *C. neoformans* com entrada no organismo através de esporos, crescimento nos pulmões com possível progressão para outras partes do corpo através do sistema circulatório (em indivíduos com deficiências imunológicas) [3].

Com a crescente consciencialização da importância clínica das infecções provocadas por fungos – com especial incidência em doentes de SIDA [10] – a comunidade científica tem vindo a dedicar mais recursos na identificação de espécies com algum grau de patogenicidade. Novas leveduras patogénicas são descobertas regularmente (com particular relevância no género *Candida*), assim como é detectado comportamento infeccioso em espécies que não eram consideradas como tal [9]. O caso da levedura *Saccharomyces cerevisiae* é disto um exemplo, dado que foi implicada em infecções em indivíduos sensíveis, quando tradicionalmente era considerado um fungo inócuo muito utilizado na indústria alimentar.

Os fungos estudados neste trabalho são: *Aspergillus fumigatus*, *Candida albicans*, *Candida glabrata*, *Candida tropicalis*, *Cryptococcus neoformans*, *Saccharomyces bayanus*, *Saccharomyces cerevisiae*, *Saccharomyces mikatae*, *Saccharomyces paradoxus* ou *Schizosaccharomyces pombe* (esta última não é considerada patogénica, mas poderá ser utilizada como controlo em experiências com amostras clínicas).

1.2. Técnicas de diagnóstico tradicionais.

As primeiras suspeitas de uma infecção por fungos são resultado da visualização clínica do paciente afectado. Posteriormente, realizam-se testes clínicos para se confirmar a sua existência e identificar o organismo responsável.

Actualmente, o diagnóstico de infecções suspeitas de terem origem em fungos patogénicos é realizado por observação directa dos tecidos afectados. Podem também ser realizadas radiografias para se analisar a evolução de micoses profundas (como por exemplo as infecções pulmonares).

Normalmente, os testes clínicos incidem sobre amostras recolhidas dos tecidos mais afectados do paciente. A observação microscópica de fragmentos de pele, fluidos corporais ou de tecidos de outros órgãos permite, em determinados casos e com grau de certeza relativamente elevado, a identificação do fungo patogénico responsável pela infecção. Esta identificação tem como base a análise da morfologia apresentada pelas células invasoras. As colónias de cada espécie (ou grupo de espécies) possui um determinado conjunto único de características estruturais. Podem apresentar grânulos de diversas formas contendo hifas, filamentos separados e finos, hifas septadas ou não septadas, diferentes tipos de células de levedura, ou esporângios, entre outros. De particular interesse é também a anotação da sua coloração, que é característica de cada espécie [6].

Frequentemente, são realizadas culturas de células, iniciadas a partir das amostras recolhidas do paciente. Estas culturas são usadas em combinação com a observação referida anteriormente com o objectivo de confirmar o diagnóstico e identificação do organismo responsável. É mesmo aconselhado que se façam sempre em paralelo para se obter um maior grau de certeza final – tendo em atenção que muitas vezes a morfologia dos fungos em cultura é diferente da que apresentam nos hospedeiros vivos.

O isolamento do espécime a cultivar é normalmente o primeiro e dos mais importantes passos destes protocolos, assim como a escolha do meio de cultivo e condições adicionais específicas. Após o tempo requerido, as culturas são examinadas macro- e microscopicamente, com a ajuda de preparações de contraste (por exemplo, preparação de hidróxido de potássio) e coloração (por exemplo, *Giema Stain* para detecção de *Histoplasma capsulatum*). Alguns testes adicionais, direccionados a determinadas espécies, podem ser realizados sobre amostras da cultura, como o *Caffeic Acid Disk Test*, que detecta

a actividade da fenoloxidase do *C. neoformans* (virtualmente o único fungo patogénico com esta propriedade) os testes enzimáticos para detectar a actividade da beta-galactosamidase e L-proline aminopeptidase da *C. albicans* [6].

Todas estas técnicas possuem um relativo grau de incerteza mesmo quando realizadas por técnicos de análises experientes, pois ligeiras modificações ao comportamento referenciado das células podem induzir em erro, com consequências potencialmente graves. Outro inconveniente é o elevado tempo necessário para se proceder à realização de algumas culturas, pois determinadas infecções têm progressões galopantes tornando-se essencial iniciar rapidamente um tratamento direccionado.

À medida que novas experiências de investigação alargam o nosso conhecimento, tornam-se necessárias mudanças nas aproximações ao diagnóstico. Com base nesta filosofia, este projecto teve como objectivo desenvolver uma metodologia alternativa de identificação de fungos patogénicos, tomando como ponto de partida a informação genética e as características químicas dos ácidos nucleicos, que permitem desenvolver sondas altamente específicas para a identificação de microrganismos ao nível da espécie, e finalmente a utilização de tecnologia avançada de *chips* de DNA.

B 2. Bases do diagnóstico molecular.

2.1. Características básicas dos ácidos nucleicos que permitem a sua utilização em diagnóstico molecular.

As células eucariótas possuem um núcleo envolvido na membrana nuclear. Dentro deste compartimento, encontra-se um componente essencial do ciclo de vida da célula - o ácido desoxirribonucleico, DNA. É este ácido nucleico, agregador de toda a informação genética de um organismo, que controla todos os mecanismos celulares, desde a replicação até à degradação celular.

A molécula de DNA é uma longa cadeia constituída por nucleótidos. Cada nucleótido é composto por uma base azotada ligada a um açúcar (2'-desoxirribose) e a uma molécula de fosfato (Figura 1-A). No DNA existem quatro diferentes tipos de bases químicas: adenina (A), guanina (G), citosina (C) e timina (T), sendo as duas primeiras classificadas como purinas e as seguintes como pirimidinas (Figura 1-B). Cada base é uma composto com ligeiras diferenças na combinação de oxigénio, carbono, azoto e hidrogénio.

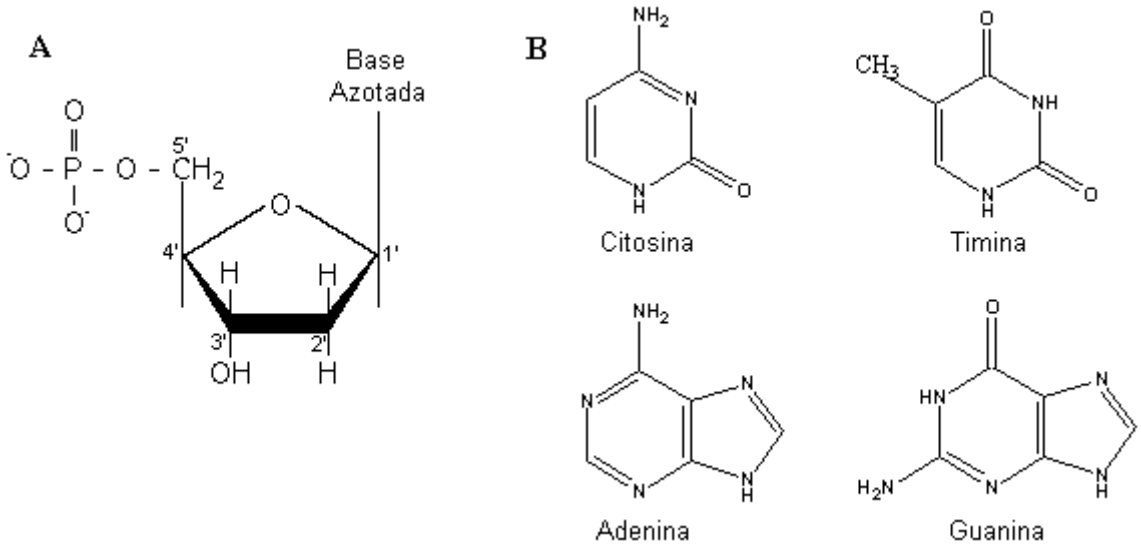

Figura 1 – A) Estrutura química de um nucleótido de DNA, constituído por um grupo fosfato, uma desoxirribose e uma base azotada. B) As quatro bases azotadas do DNA. Adaptado de http://www.uq.edu.au/vdu e http://web-mcb.agr.ehime-u.ac.jp.

Os nucleótidos de uma cadeia de DNA estão unidos sequencialmente uns aos outros, com o (carbono 3' do) açúcar de um ligado ao (oxigénio do) grupo fosfato do seguinte, através de uma ligação fosfodiéster. A ordem final em que estes constituintes

básicos estão alinhados contém a informação que a molécula de DNA utiliza para controlar a célula. Esta sequência possui uma orientação, ou sentido de leitura, dado que o término dos seus dois extremos é quimicamente diferente: um termina com um grupo 5'-fosfato, enquanto que o outro finaliza com um grupo 3'-hidroxil, sendo designados terminal 5' e terminal 3', respectivamente. Esta orientação é muito importante em todos os processos em que o DNA está envolvido – por exemplo, a síntese de novas cadeias faz-se sempre no sentido 5' para 3' [11].

A estrutura do DNA, como foi determinada por *Watson* e *Crick* em 1954, apresenta-se em forma de uma molécula de dupla cadeia consistindo em dois polinucleótidos orientados em sentidos opostos (cadeias anti-paralelas) que se enrolam em torno de um eixo comum. Nesta estrutura, os dois eixos de açúcar e fosfato encontram-se expostos ao exterior (maior estabilidade hidrofílica) enquanto que as bases, viradas para dentro, formam ligações de hidrogénio. As bases emparelham com a sua complementar, sempre segundo o mesmo padrão: adenina emparelha com timina e guanina com citosina. O primeiro par estabelece duas pontes de hidrogénio entre si e o segundo três ligações do mesmo tipo (Figura 2). As ligações por pontes de hidrogénio são atracções electrostáticas relativamente fracas entre um átomo electronegativo (como o oxigénio ou o azoto) e um átomo de hidrogénio ligado a outro átomo electronegativo [12].

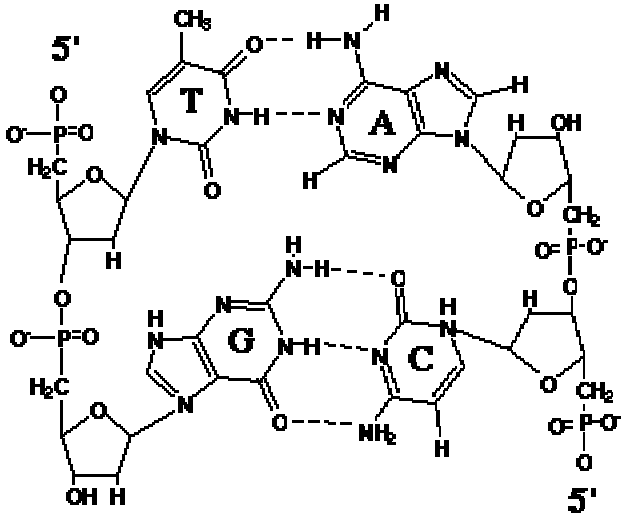

Figura 2 – Ligações de hidrogénio entre bases azotadas complementares de duas cadeias anti-paralelas. Adaptado de http://www.blc.arizona.edu.

A natureza química das bases deste dímero cria um ligeira força de torção que concede à dupla cadeia de DNA a sua estrutura característica de dupla hélice – com uma rotação característica da mão direita (*right-handed*). Como uma purina (adenina e guanina) emparelha sempre com uma pirimidina (timina e citosina), a forma final de cada par de bases é muito semelhante. Isto permite que a estrutura em dupla hélice seja muito homogénea. As bases posicionam-se quase perpendicularmente ao eixo comum e bases adjacentes são separadas por 3.4 Å. A estrutura helical repete-se a cada 34 Å, logo há dez bases por cada volta (34 Å /3.4 Å) e por cada base existe uma rotação de 36 graus (360º/10 bases). O diâmetro total da dupla hélice de DNA é de 20 Å [13]. Devido à configuração desta molécula de DNA, no seu exterior são visíveis duas "ranhuras" de dimensões diferentes que acompanham a hélice em toda a extensão da sua espiral. A mais larga e profunda é designada por *major groove*. A outra, mais estreita e de menor relevo é a *minor groove*. Por conseguinte, todas as bases estão acessíveis ao exterior da hélice para contactarem com outras moléculas, como as proteínas que interagem com o DNA, sem que seja necessário romper as ligações da dupla cadeia (embora, caso seja necessário, possa ocorrer distorção local da hélice) [12].

A molécula descrita anteriormente é designada por B-DNA, sendo a forma que aparece predominantemente nas células vivas, mas o DNA de dupla cadeia pode adoptar diferentes configurações. A forma A-DNA (constitui-se em condições de muito baixa humidade) tem 11 bases por volta, é mais compacta, apresenta ranhuras diferentes e as bases encontram-se em ângulos diversos. Ao contrário das configurações A e B, a forma Z-DNA possui uma rotação característica da mão esquerda (*left-handed*) e uma hélice mais irregular.

Para além das pontes de hidrogénio, existem ainda outros tipos de ligação envolvidos na estabilização na dupla hélice. Entre alguns pares de átomos das bases ocorre atracção através de forças de *Van der Walls*. Estas forças, geralmente muito fracas, são potenciadas pelo elevado número de átomos que interagem desta forma. Estabelecem-se igualmente interacções hidrofóbicas ("interacções π-π") entre pares de bases adjacentes, resultando na exposição ao exterior (moléculas de água) das superfícies mais polares. Estas interacções ajudam a estabilizar a estrutura do DNA, após ocorrer o emparelhamento das cadeias.

Devido ao padrão de emparelhamento dos pares de bases (adenina com timina e guanina com citosina), a sequência de cada cadeia do DNA celular é complementar à sequência da outra, podendo servir-lhe de molde. Durante a replicação do DNA ocorre a separação das duas cadeias, e cada uma irá dar a origem a uma nova dupla hélice idêntica à original. Neste processo, dito semi-conservativo, duas novas cadeias de DNA são formadas (cada uma copiada a partir de uma das cadeias originais). Cada uma das duas novas duplas hélices é constituída por uma cadeia original e uma nova.

Para além do DNA, existe outro ácido nucleico com um papel essencial no metabolismo da célula: o ácido ribonucleico (RNA). Esta molécula é também um polinucleótido, mas com duas diferenças principais relativamente ao DNA: os seus nucleótidos possuem como açúcar uma ribose (com mais um grupo hidroxil, ligado ao carbono da posição 2') e tem uma base diferente que substitui a timina – o uracilo (U). O RNA é, na maioria dos casos, uma molécula de cadeia única mas, tal como o DNA, apresenta a cadeia de nucleótidos ordenada no sentido de leitura 5' para 3'. Uma molécula de RNA celular pode ter uma extensão variável mas, regra geral, apenas atinge alguns milhares de nucleótidos, ao contrário do DNA celular, que pode atingir milhões de pares de bases (bp, unidade de medida da extensão das molécula de dupla cadeia). As funções do RNA estão intimamente associadas às do DNA, e serão referidas na secção 2.3.

Ao contrário do DNA, que existe principalmente como estrutura tridimensional em dupla hélice, o RNA de cadeia única pode assumir várias conformações. As suas estruturas secundárias mais simples envolvem o emparelhamento entre pares de nucleótidos complementares da mesma cadeia. O RNA pode também existir sob a forma de estruturas tridimensionais secundárias (ou mesmo terciárias) mais complexas. As várias conformações permitem obter o máximo de estabilidade termodinâmica da molécula e estão relacionadas com a função que cada RNA exerce na célula [11].

2.2. Hibridação de ácidos nucleicos complementares em diagnóstico molecular.

Durante a replicação de DNA, as duas cadeias da dupla hélice separam-se uma da outra. Este processo celular envolve enzimas (helicases) que consomem energia química na forma de ATP. Em experiências laboratoriais, este processo pode ser simulado aquecendo uma solução contendo fragmentos de DNA. O calor provoca o rompimento das ligações de

hidrogénio, que unem os pares de bases, e consequentemente a dupla hélice é dissociada. Este fenómeno associado ao aquecimento, designa-se por fusão (ou desnaturação), porque ocorre de forma relativamente abrupta e a determinada temperatura. A temperatura a que metade da estrutura da dupla hélice de DNA se funde é designada por temperatura de fusão (Tm na sigla em inglês). Para além do calor, as cadeias podem também ser separadas por adição de soluções alcalinas ou de agentes desnaturantes à solução de DNA, que ionizam as bases dos nucleótidos e quebram as ligações [13].

Se a temperatura se mantiver abaixo da Tm do ácido nucleico em solução, as cadeias complementares voltam a associar-se espontaneamente (renaturação) formando uma nova dupla hélice idêntica à que existia antes do aquecimento. Esta elasticidade da molécula de DNA é crucial para as funções biológicas em que está envolvido na célula e pode ser usada em diagnóstico molecular.

De referir que o RNA (embora na maioria dos casos se apresente sob a forma de cadeia única) também possui a capacidade de formar dupla cadeia RNA-RNA e mesmo RNA-DNA, com características semelhantes às descritas para as duplas cadeias DNA-DNA.

Numa experiência laboratorial, o fenómeno de emparelhamento de dois polinucleótidos através do estabelecimento de ligações de hidrogénio entre pares de bases de cada um é designado por hibridação. Como a palavra indica, ocorre a formação de uma dupla cadeia a partir de ácidos nucleicos de origens diferentes (uma estrutura "híbrida"). Esta é a base de várias técnicas laboratoriais utilizadas para estudar o relacionamento entre duas amostras de DNA ou para detectar e isolar moléculas específicas de DNA numa mistura contendo diferentes sequências [11]. Normalmente as experiências de hibridação de ácidos nucleicos envolvem a utilização de sondas – moléculas polinucleotídicas de sequência conhecida – que são usadas para detectar, por emparelhamento, outras moléculas – os "alvos" – dentro de uma mistura heterógenea de vários fragmentos de ácidos nucleicos. A sonda e o alvo estão relacionados através da sequência, podendo ser totalmente complementares ou não [14].

Alguns técnicas que se baseiam na hibridação são a Reacção em cadeia da polimerase (PCR, sigla em inglês para *Polimerase Chain Reaction*) e os *microarrays* de DNA, também designados por *chips* de DNA (secção 3).

Existem alguns aspectos importantes a ter em conta quando se planeia a elaboração de uma experiência de hibridação de ácidos nucleicos. Entre eles, a temperatura de fusão e a formação indesejada de estrutura secundária nas cadeias simples. É necessário controlar rigorosamente estes dois factores para que a hibridação ocorra na extensão desejada.

2.2.1. Temperatura de fusão.

A Tm a que duas cadeias complementares de DNA se dissociam (e reassociam) depende de vários factores, sendo o comprimento da dupla cadeia o mais determinante. Cadeias longas, com grande número de ligações de hidrogénio necessitam de mais energia térmica para serem separadas; mas, acima de um certo comprimento (aprox. 500 bases), este efeito é negligenciável. A composição em bases é igualmente um importante factor que afecta a temperatura de fusão. Moléculas com grande concentração de pares de guanina com citosina (conteúdo G+C) requerem temperaturas mais altas para desnaturar porque o emparelhamento G≡C (três ligações de hidrogénio) é mais estável do que o dos pares adenina com timina (duas ligações de hidrogénio). O ambiente químico em que a amostra de DNA se encontra é outro factor que influencia a Tm. A presença de catiões monovalentes (por exemplo, iões Na^+) estabiliza a dupla hélice – aumenta a Tm – enquanto que desnaturantes químicos, como a ureia e a formamida têm um efeito exactamente oposto [14]. Se houverem pares de bases que não emparelham (por exemplo A e C) a Tm diminui, porque a disrupção do emparelhamento cria falhas na estrutura de dupla hélice, causando instabilidade na estrutura e é necessária menor energia para a desnaturar.

A temperatura de fusão (Tm) é o valor ao qual se estabelece um equilíbrio dinâmico entre as estruturas em dupla cadeia e moléculas desnaturadas em cadeia simples. É por este motivo que a Tm é importante tanto para o processo de desnaturação/fusão como para o de hibridação. Deste modo, para que duas cadeias complementares hibridem, a temperatura da experiência tem que ser inferior à Tm.

Quando se procura obter hibridação apenas entre cadeias totalmente complementares, a temperatura da experiência é mantida apenas 4-5 ºC abaixo da Tm das hélices perfeitas – as únicas que se mantêm estáveis no fim da experiência. Se a temperatura da experiência for substancialmente inferior, ocorrerá a formação adicional de hélices imperfeitas (com pares de bases que não emparelham), entre cadeias que não são

totalmente complementares. Estas condições são utilizadas, por exemplo, para identificar genes da mesma família – com sequências parcialmente idênticas [15].

O cálculo da temperatura de fusão é realizado por fórmulas que fornecem aproximações ao seu valor real, já que este apenas pode ser exactamente determinado na prática. O método básico baseia-se apenas na composição em bases e fornece uma indicação grosseira da Tm para pequenos oligonucleótidos (15-30 nucleótidos de extensão). A sua fórmula é a seguinte : *Tm = (4 x número de G + C) + (2 x número de A + T)*, em que G, C, A, T correspondem aos nucleótidos guanina, citosina, adenina e timina [12].

O cálculo da Tm pode ser feito com recurso a fórmulas mais adaptadas às condições específicas da hibridação. Ajustando a fórmula básica à concentração de sais, obtém-se a seguinte equação para as duplas cadeias DNA-DNA:

$$Tm = 81.5 + 16.6 (\log_{10}[Na^+]) + 0.41 (\%GC) – 500/L$$

$[Na^+]$ representa a concentração de catiões monovalentes em solução, %GC é a percentagem das bases guanina e citosina e "L" é a extensão da dupla cadeia em pares de bases. Esta fórmula está limitada a concentrações de catiões monovalentes entre 0.01 e 0.4 M e a percentagens G+C entre 30 e 75% [14].

A Tm de sequências longas pode ser calculada recorrendo a métodos termodinâmicos baseados na aplicação do modelo *Nearest-Neighbour* aos ácidos nucleicos. Este modelo assume que a estabilidade de uma dupla hélice de DNA depende da identidade e orientação de pares de bases "vizinhos". Existem dez interacções *Nearest-Neighbour* diferentes nesta molécula correspondendo às possíveis combinações de dois pares de bases das duas cadeias: AA/TT, AT/TA, TA/AT, CA/GT, GT/CA, CT/GA, GA/CT, CG/GC, GC/CG e GG/CC (o "/" separa cadeias com orientação antiparalela). A cada dupleto de pares de bases são atribuídos valores de energia livre, ΔG (representa a sua estabilidade relativa), de entalpia, ΔH (calor libertado ou absorvido pela molécula) e de entropia, ΔS (medida da aleatoriedade da molécula). Com base nesses valores [16], na contagem de todos os dupletos presentes e aplicando equações termodinâmicas é possível determinar a estabilidade total da dupla hélice, bem como a sua Tm [17-19] (ver Anexo 9). Os cálculos realizados segundo estes métodos possuem adicionalmente como variáveis a concentração das cadeias de oligonucleótidos e a concentração de sais na solução. Para

diferentes tipos de cadeias (DNA-DNA, RNA-RNA ou DNA-RNA) são realizados ajustes nas fórmulas a aplicar.

2.2.2. Estrutura secundária.

No início de uma experiência de hibridação, os ácidos nucleicos de interesse (sondas, "alvos" ou outros) têm que estar na forma de cadeia única, para que possa ocorrer o emparelhamento entre moléculas de origem diferente. Como foi referido para o RNA, os ácidos nucleicos de cadeia simples têm tendência a adoptar uma estrutura secundária (entre si ou com outras moléculas) que lhes permita minimizar termodinamicamente as repulsões electrostáticas/hidrofóbicas do meio circundante.

Quando duas cadeias emparelham para formar uma dupla hélice, há uma perda total de entropia pois a estrutura resultante é mais rígida do que as cadeias simples. No entanto, a este factor desfavorável sobrepõe-se o decréscimo de energia livre decorrente do emparelhamento das bases. Cada par de bases adicional na estrutura promove a diminuição da energia livre pelo estabelecimento das ligações de hidrogénio entre si e pelas interacções estabilizadoras de *Van der Walls* e hidrofóbicas com o par adjacente. Se o valor total da energia livre atingir um nível mínimo, a reacção de hibridação ocorre espontaneamente [20]. O mesmo princípio é aplicado ao processo de formação de estrutura secundária de uma só cadeia de ácidos nucleicos.

Uma cadeia de DNA que possua na sua sequência duas zonas complementares (em sentidos inversos da cadeia) e relativamente próximas dobra-se sobre si mesma com o emparelhamento dessas bases – por ligações de hidrogénio iguais às da dupla hélice. A estrutura conseguida pode variar mediante a distância entre as duas zonas, o número de bases complementares e a ocorrência de *mismatches* (pares de bases que não emparelham) ou *unmatches* (nucleótido sem par numa das duas zonas complementares). As conformações mais simples incluem o gancho (*hairpin*) (Figura 3) que se forma quando as bases complementares se encontram muito próximas (aproximadamente 5-10 nucleótidos) e o haste-laço (*stem-loop*), para bases mais distantes (entre 50 e várias centenas de nucleótidos). A associação destas conformações básicas, pode originar estruturas terciárias mais complexas, como o pseudo-nó [13].

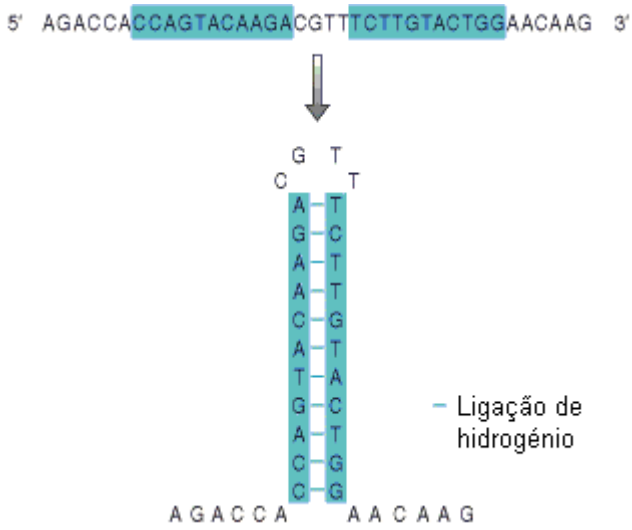

Figura 3 – Formação de um gancho numa cadeia simples de DNA. Adaptado de Strachan T, 1999, Oxford [14].

A ocorrência deste fenómeno numa experiência de hibridação deve ser minimizada e se possível evitada porque as bases que se encontram associadas com outras da mesma cadeia não estão disponíveis para mais ligações, estando certas regiões da conformação secundária (ou terciária) posicionalmente bloqueadas dentro da própria estrutura. Desta forma, a cadeia não hibrida com outra(s) ou apenas o consegue parcialmente e de forma instável, afectando o rendimento final e a especificidade da experiência. Para minimizar a formação de estruturas secundárias em ácidos nucleicos de cadeia simples, devem ser escolhidas moléculas que não possuam zonas complementares na sua sequência primária ou sejam o mais pequenas possíveis.

2.3. Genes e o "Dogma Central".

Como foi referido, é na sequência de bases do ácido desoxirribonucleico que se encontra toda a informação genética da célula. Esta sequência é igual em todas as células do mesmo organismo e controla todos os mecanismos celulares, efectuados na sua maioria por proteínas e RNA's.

O fundamento principal da biologia molecular, também designado por "Dogma Central" declara que a informação genética flui do DNA para o RNA e por fim para as proteínas. Cada segmento de DNA que dá origem a uma proteína é designado por gene. As transferências da informação do DNA para o RNA e do RNA para a proteína são realizados por processos fiáveis e com elevado grau de exactidão. De destacar, que nestes

processos intervêm três tipos diferentes de RNA: RNA mensageiro (mRNA), RNA ribossomal (rRNA) e RNA de transferência (tRNA).

A síntese de mRNA a partir do DNA é designada por transcrição e é catalisada por uma enzima da família das RNA polimerases. Nos procariótas apenas é usada uma enzima para a transcrição de todos os genes enquanto que nos eucariótas existem três tipos (por exemplo, a RNA polimerase II é usada na transcrição de todos os genes codificadores de proteínas). Estas enzimas reconhecem os locais (promotores), que antecedem a sequência do gene, onde se devem ligar ao DNA e iniciar a síntese. Para que isto possa ocorrer as duas cadeias de DNA terão que se ter separado localmente com a ajuda de enzimas. A síntese de uma cadeia de mRNA faz-se com a adição, ao terminal 3', de nucleótidos complementares à sequência de uma das cadeias do DNA (*antisense* ou não codificante). A transcrição termina quando a polimerase atinge um sinal de terminação. Forma-se deste modo um mRNA com sequência igual à da cadeia codificante (*sense*) do DNA, com as timinas substituídas por uracilos.

Enquanto que nas células procariótas o ácido ribonucleico recém formado é um mRNA funcional, nas células eucariótas ainda terá que passar por processos de maturação. Os terminais 5' e 3' são quimicamente modificados, porções da sua sequência (intrões) são retirados e os restantes (exões) são unidos – processo de *splicing*. Este processamento do mRNA pode não ser sempre realizado da mesma forma, dando origem a diferentes combinações de exões, logo diferentes proteínas – *splicing* alternativo. O mRNA atinge finalmente o citoplasma (nos eucariótas), onde vai ser recrutado por ribossomas dando início à tradução.

Os ribossomas são pequenos complexos RNA-proteína constituídos por duas subunidades de tamanhos diferentes. Cada subunidade possui várias proteínas distintas e moléculas de rRNA de diferentes tamanhos. Nos eucariótas, a subunidade pequena, 40S ("S" refere-se a unidades de *Svedberg*, uma medida da taxa de sedimentação durante a centrifugação), contém uma cadeia com cerca de 1900 nucleótidos denominada 18S enquanto que a subunidade grande, 60S, possui três cadeias de rRNA: a 28S (aproximadamente 4800 nucleótidos), a 5.8S (apróx. 160) e a 5S (apróx. 120). A união das duas subunidades origina a estrutura funcional designada por 80S. Nos procariotas a constituição é semelhante mas com pequenas diferenças no tamanho das cadeias e a

inexistência de um correspondente à cadeia 5.8S. O seu complexo ribossomal completo, ligeiramente mais pequeno é designado por 70S.

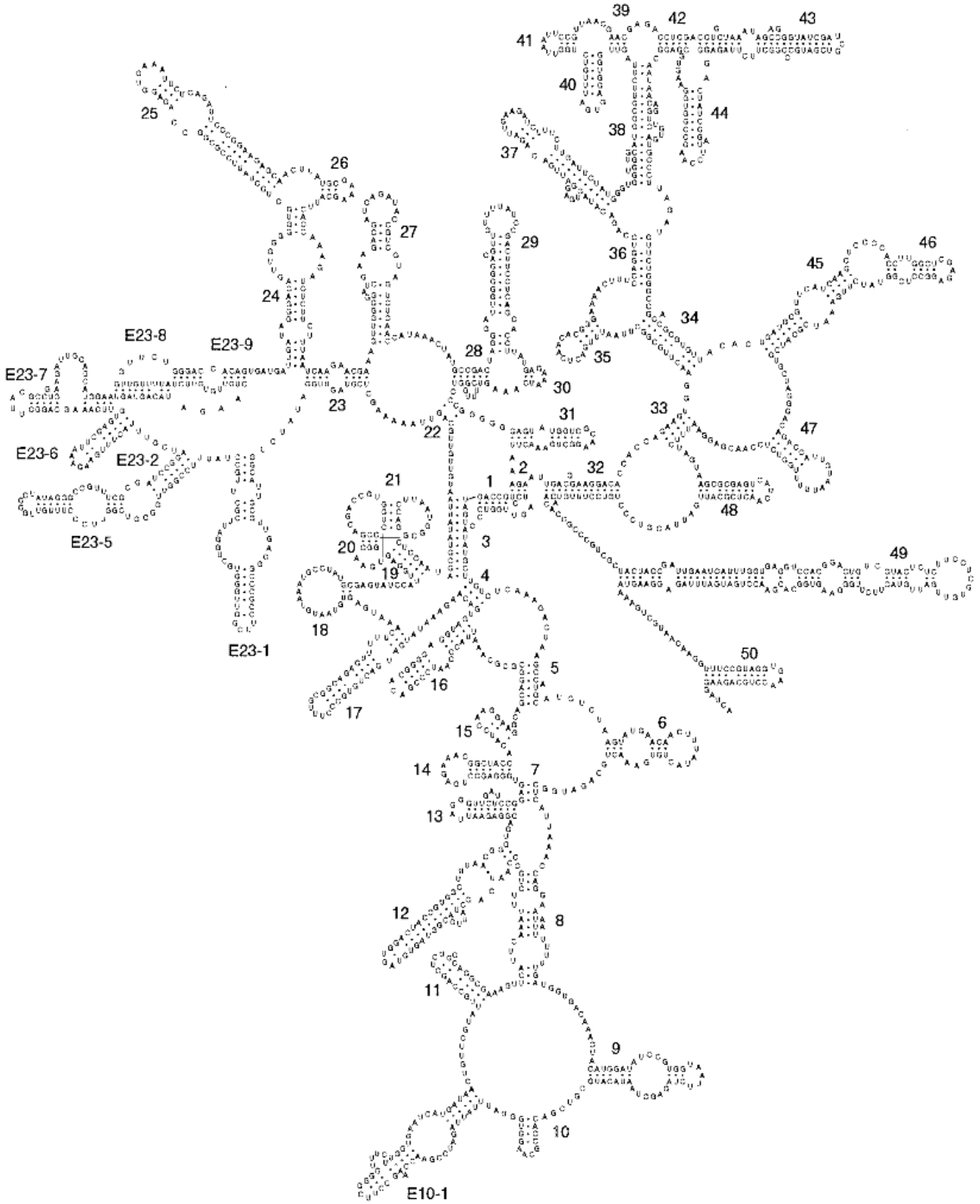

Figura 4 – Estrutura secundária da cadeia 18S do RNA ribossomal da espécie de alga *Chlorarachnion reptans*. Adaptado de Van de Peer, 1997 [21].

As cadeias de rRNA possuem uma estrutura secundária muito complexa (especialmente a 18S e a 28S) o que influencia a conformação tridimensional do ribossoma e por conseguinte a sua função. Embora a sequência primária das suas cadeias varie consideravelmente entre espécies diferentes (ver secção 2.5) as estruturas secundárias resultantes são, na maioria dos casos, similares. Existe a formação de várias estruturas em haste-laço com a obtenção de um máximo de bases emparelhadas (Figura 4).

O mRNA é usado como molde para a síntese proteica que é realizada segundo um código genético universal, com pequenas adaptações em determinadas espécies. A cada conjunto sequencial de três nucleótidos do mRNA, designado por codão, corresponde um aminoácido a ser inserido na sequência primária da proteína. Alguns codões, porém, têm funções unicamente operacionais.

2.3.1. Genes do rRNA.

Os genes que codificam as cadeias 18S, 5.8S e 28S (25S em algumas espécies) do RNA ribossomal encontram-se dispostas sequencialmente no DNA (Figura 5) apresentando a mesma configuração em todas as espécies eucariótas. Entre o gene do 18S e o do 5.8S e entre este e o do 28S encontram-se duas regiões também transcritas, designadas por *Internal Transcribed Spacers* (ITS1 e ITS2). Este agrupamento de genes encontra-se repetido múltiplas vezes no DNA de uma célula, permitindo a rápida síntese de grandes quantidades de rRNA.

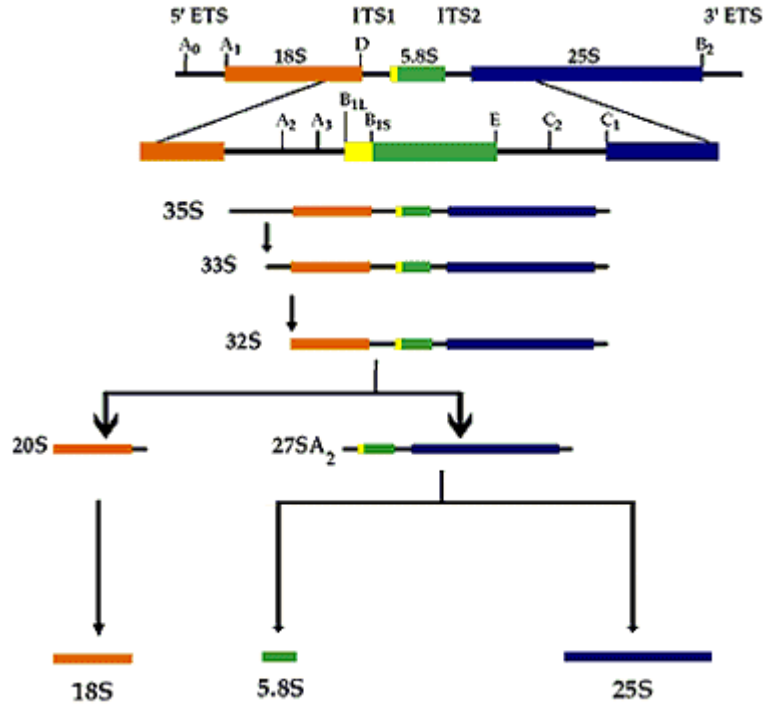

Figura 5 – Transcrição dos genes de rRNA em *S. cerevisiae*. O transcrito primário é processado, com formação final das cadeias 18S, 5.8S e 25S. Adaptado de http://www.bio.cmu.edu.

Nos eucariótas, na transcrição do rDNA (DNA que codifica rRNA) é utilizada a RNA polimerase I, ocorrendo a síntese de uma única molécula de pré-RNA – o transcrito primário. Este contém as regiões 18S, ITS1, 5.8S, ITS2 e 28S e zonas adjacentes nos terminais 5' e 3'. O processamento do transcrito primário é realizado no nucléolo e conduz à formação das subunidades 40S e 60S. No primeiro passo as proteínas ribossomais associam-se ao pré-RNA. De seguida decorre uma série de clivagens que separam o transcrito primário nos seus três constituintes (18S, 5.8S e 28S) e removem os dois ITS bem como as regiões terminais. As subunidades ribossomais pequena (com o rRNA 18S) e grande (com o 28S associado ao 5.8S) são finalmente enviadas para o citoplasma na sua forma definitiva [22].

À subunidade 60S é também associado o rRNA 5S. Esta cadeia tem uma origem diferente das anteriores, pois é codificada em regiões distantes do DNA. A sua síntese não é feita em coordenação com os outros rRNA e é coadjuvada pela RNA polimerase III.

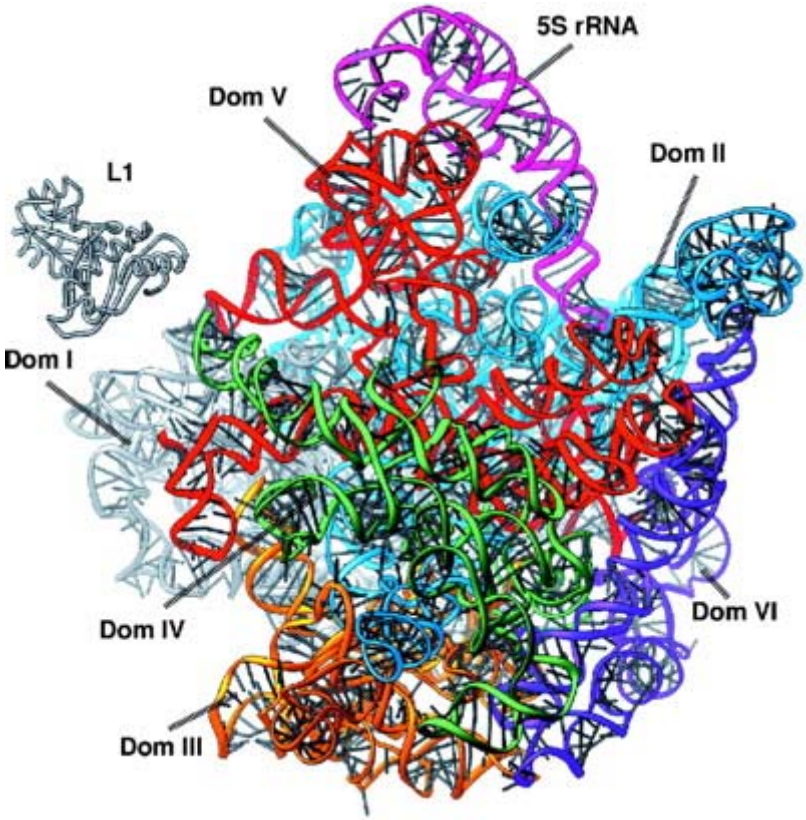

Figura 6 – Estrutura terciária das cadeias de rRNA da subunidade grande do ribossoma de *Haloarcula marismortui*. Cada domínio (Dom) numerado da cadeia 23S foi representado com uma cor diferente. A cadeia 5S é visível na parte superior da figura. Adaptado de Ban, 2000 [23].

2.4. Variabilidade dos genomas e o diagnóstico molecular.

O genoma de um organismo vivo é constituído por toda a informação genética inscrita na sequência de DNA existente em cada uma das suas células. Isto significa que engloba todos os genes (codificadores de mRNA/proteína, rRNA ou tRNA), mas também regiões intergénicas não codificantes. Estas regiões contêm sequências reguladoras (por exemplo, de transcrição ou repressão), sequências repetidas e outros segmentos diversos dos quais alguns ainda não se sabe a função. As sequências repetidas podem ser distribuídas ao longo do genoma de modo aparentemente aleatório ou dispostas consecutivamente numa única região.

Uma grande diferença entre os genomas eucariótas encontra-se no tamanho. Os genomas mais pequenos são da ordem dos 10 Mb (1 Mb = 1 000 kb = 1 000 000 bp), enquanto nos maiores esse número pode atingir os 100 000 Mb. O número de genes também apresenta grande variabilidade entre espécies diferentes. Estas duas características aparentam estar vagamente relacionadas com o grau de complexidade do organismo. Eucariótas superiores, como os animais vertebrados e as plantas angiospérmicas, possuem,

regra geral, genomas maiores e maior número de genes, ao contrário das eucariótas mais simples, como os fungos. O genoma do homem (*Homo sapiens*) possui entre 30 000 e 40 000 genes nas suas 3200 Mb [24, 25]. A planta *Arabidopsis thaliana* com um genoma constituído por 125 Mb possui cerca de 25 000 genes [26]. O nemátodo *Caenorhabditis elegans* tem 19 000 genes num genoma de 97 Mb [27]. A levedura *Saccharomyces cerevisiae* tem, por seu lado, um pequeno genoma com 12.1 Mb de extensão dividido por 16 cromossomas e 5800 genes [28].

Como se constata, não existe um relação proporcional entre tamanho e número de genes, o que se explica por uma maior concentração de zonas codificantes nos genomas mais pequenos. Geralmente, estes possuem menos intrões (e mais pequenos) e menor número de repetições distribuídas pelo genoma. Estas sequências repetidas que, no homem, representam cerca de 44% da totalidade do genoma, são uma das principais causas da grande extensão dos genomas nos eucariótas superiores [12].

Os genomas procariótas são muito diferentes dos eucariótas, começando pela forma e tamanho. Aqueles genomas encontram-se essencialmente numa única molécula circular de DNA (sem terminais) e são geralmente pequenos em extensão e em número de genes. Como exemplo, o genoma de *Escherichia coli* K12 possui apenas 4639 kb e 4405 genes [29]. Os procariótas possuem ainda moléculas circulares ou lineares de DNA adicionais designados por plasmídeos. O seu tamanho e importância nos mecanismos celulares é muito variado, contribuindo para dificultar o estabelecimento de características comuns a todos os genomas procariótas. Um princípio comum é a grande concentração de regiões codificantes existente nestes genomas, com zonas intergénicas mínimas [29] e inexistência de intrões.

Algumas espécies de vírus apresentam um caso especial de genoma, pois os seus genes estão codificados, não em DNA, mas em moléculas de RNA.

Os genomas são entidades dinâmicas que se modificam ao longo do tempo em resultado do efeito cumulativo de alterações na sua sequência provocadas por mutações ou recombinação genética.

Uma mutação é uma alteração numa pequena sequência do genoma. Muitas mutações são apenas substituições de um nucleótido por outro, enquanto que noutros casos resulta da inserção ou eliminação de um número reduzido de nucleótidos. Erros na

replicação do DNA ou a acção de agentes mutagénicos (reagem com o DNA e alteram a sua estrutura) estão na origem da grande maioria das mutações. Todas as células possuem mecanismos de reparação que conseguem minimizar a ocorrência deste fenómeno pela acção de enzimas. Em certos casos, porém, estes mecanismos são incapazes de conservar a estrutura original do DNA e permitem que um genoma preserve a mutação [12].

A recombinação resulta numa reestruturação de uma parte significativa do genoma. Pode resultar dos processos naturais de troca de cadeias de DNA entre cromossomas homólogos durante a meiose ou da transposição de uma sequência dentro do mesmo cromossoma ou entre cromossomas diferentes. A recombinação é um processo celular que decorre sob a acção e regulação de certas proteínas. A sua ocorrência permite aumentar a variabilidade dentro da mesma espécie.

Os efeitos das mutações ou da recombinação podem ser diversos. A mutação de um gene essencial pode resultar na morte da célula, caso a proteína que ele codifique deixe de ser viável. Por outro lado, em raríssimos casos, um gene mutado pode ser benéfico para a célula se a proteína expressa lhe proporcionar uma característica vantajosa. Noutros casos a ocorrência deste fenómeno têm um impacto insignificante ou mesmo nulo. As mesmas consequências imprevisíveis podem ser observadas como resultado de recombinação genética, embora com menor incidência de casos deletérios, já que é um processo natural e promovido pelos mecanismos celulares.

Todas as alterações genéticas não letais têm o potencial para contribuir para a evolução do genoma em que ocorrem. No entanto, esse potencial só é efectivo se a mutação ou recombinação forem transmitidas aos descendentes. Em células unicelulares, este não é um factor condicionante, pois as células filhas irão herdar todo o património genético da célula mãe. Em organismos multicelulares, apenas são transmitidas as novidades genéticas existentes nas células reprodutivas, anulando-se evolutivamente as restantes [12].

A acumulação destas modificações numa dada população pode ter uma magnitude tal que o genoma comum a todos os seus organismos seja claramente distinto do genoma de outras populações anteriormente idênticas. Nestes casos, deu-se a formação de uma nova espécie independente com traços relativamente semelhantes aos da sua ancestral.

2.5. Identificação molecular de organismos baseada em relações de filogenia.

O agrupamento de espécies ou taxonomia tem sido uma preocupação constante dos investigadores desde que Lineu, no século XVIII, dividiu os seres de acordo com as semelhanças e diferenças físicas. Estas classificações que eram originalmente desenvolvidas à luz das teorias criacionistas, foram adaptadas à teoria da evolução de Darwin. Espécies com aspecto idêntico teriam, portanto, um ancestral comum que seria tanto mais recente quanto maiores fossem as semelhanças entre elas. Este tipo de representação evolutiva baseado nas características dos organismos é designado por filogenia.

Com o decorrer do tempo e com a propagação de novas metodologias científicas, os dados a ter em conta no estabelecimento de relações filogenéticas foi-se alterando. Actualmente, mais do que os aspectos morfológicos, são tidos em conta as propriedades moleculares das espécies em estudo, como as sequências de proteínas e de ácidos nucleicos. Estes dados, têm a vantagem de não serem ambíguos, poderem constituir bases de dados enormes mas compreensíveis e serem facilmente analisados por métodos matemáticos. Este tipo de filogenia, molecular, tem originado resultados por vezes surpreendentes e contraditórios às aproximações morfológicas tradicionais.

A importância do DNA nos estudos filogenéticos é relativamente maior do que a das proteínas, porque as sequências de ácidos nucleicos possuem mais informação evolutiva do que as sequências de aminoácidos. Se, por exemplo, ocorrer mutação num nucleótido de um determinado codão, o aminoácido que ele codifica não é alterado desde que o novo codão seja sinónimo do inicial. Deste modo, a análise das regiões codificantes e não codificantes dos genomas permite a obtenção de grande quantidade de informação acerca da evolução das espécies.

O objectivo da maioria das análises filogenéticas é construir diagramas em forma de árvore que descrevam visualmente as relações evolutivas das espécies em estudo. A árvore da Figura 7 é uma representação das hipotéticas relações filogenéticas entre o homem, o ratinho e a mosca. Cada nódulo representa uma unidade taxonómica (espécies, populações, indivíduos). Os nódulos externos e internos identificam as unidades taxonómicas em estudo (homem, ratinho e mosca, nesta árvore) e as ancestrais, respectivamente. Os ramos definem as relação entre as diferentes unidades taxonómicas

(descendente ou ancestral). O comprimento de cada ramo representa o número de alterações que ele sofreu. Normalmente as árvores com ramos de comprimento variável estão calibradas para representar a passagem do tempo. A raiz é o ancestral comum a todas as espécies em análise [12, 30].

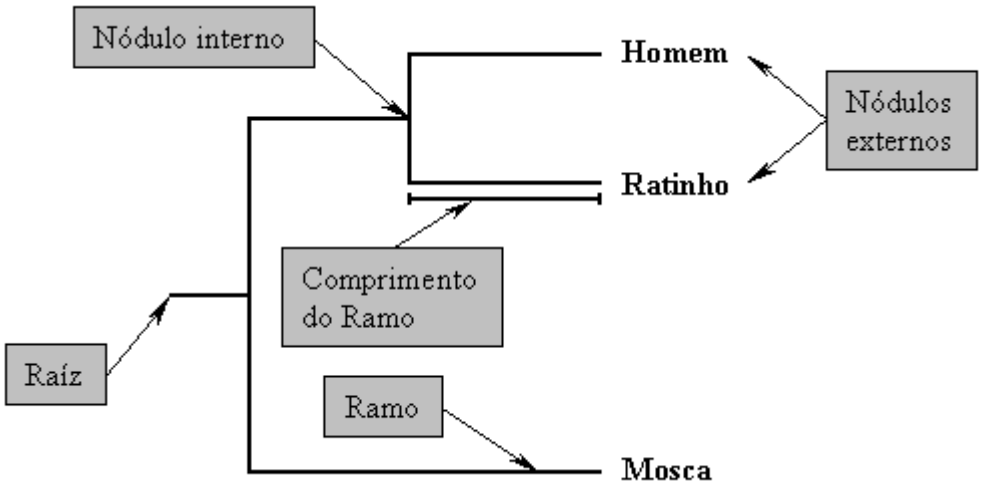

Figura 7 – Hipotética árvore filogenética para as espécies homem, ratinho e mosca. Os elementos essenciais destes diagramas são os nódulos, os ramos e a raiz, que representam as unidades taxonómicas, as suas relações filogenéticas e o ancestral comum, respectivamente.

Existem outras configurações possíveis para representar uma árvore filogenética, nomeadamente aquelas em que todos os ramos têm o mesmo comprimento ou as que não possuem uma raiz. Neste último caso, apenas estão especificadas as relações entre espécies, sem se identificar um ancestral comum ou linha evolutiva.

Para além das árvores filogenéticas que têm como objecto um conjunto vasto de características das espécies, é também possível construir árvores referentes apenas a um gene ou conjunto limitado de genes. Nos estudos que conduzem a estas representações são tidos em conta os genes ortólogos, ou seja, genes com um ancestral evolutivo comum (deduzido pelas semelhanças nas sequências) e que pertencem a espécies diferentes. A sequência ancestral comum aos genes ortólogos antecede no tempo a divisão que ocorreu entre as espécies. Fazendo a comparação entre estes genes é possível inferir a relação filogenética entre as espécies das quais os genes foram obtidos. No entanto, é necessário ter em atenção que a separação entre genes diferentes através da alteração das sequências genéticas muito provavelmente não terá ocorrido exactamente na mesma altura que a

divisão das espécies (especiação), um fenómeno mais complexo que depende de outros factores [30].

Uma árvore filogenética baseada nas sequências de DNA é construída com base em resultados de alinhamentos múltiplos (secção 4.3). As sequências dos genes ortólogos em estudo são dispostos lado a lado de modo a que os nucleótidos homólogos possam ser comparados. Para proceder a estes alinhamentos são utilizados programas de computador capazes de analisar múltiplos genes de uma só vez. Dependendo do método utilizado, o alinhamento múltiplo de sequências é convertido em valores numéricos capazes de serem analisados matematicamente [30]. Com base nestas análises de valores, é finalmente construída a árvore filogenética que representa as relações evolutivas entre os genes das diferentes espécies. Sobre esta árvore são realizados testes que deduzem o seu grau de fiabilidade e pode ainda ser adicionada informação acerca da variável tempo, através do comprimento dos seus ramos (cronómetro molecular).

Um grupo de genes sobre o qual têm incidido vários estudos filogenéticos é o das sequências codificantes do RNA ribossomal (secção 2.3.1). Estes têm sido utilizados na definição da taxonomia e filogenia de uma grande variedade de espécies, procariotas ou eucariótas.

Estudos filogenéticos com os genes das cadeias de rRNA permitiram corroborar e levantar dúvidas acerca da classificação evolutiva de certos géneros e espécies de leveduras, anteriormente estabelecida com recurso a dados morfológicos.

A análise de sequências dos genes da cadeia 18S do rRNA permitiram verificar a heterogenicidade filogenética de várias espécies do género *Kluyveromyces* e a grande homogeneidade dos géneros *Brettanomyces* e *Dekkera* [31]. Nesse estudo, foi igualmente determinado que as espécies *Debaromyces castelli*, *Debaromyces hansenii* e *Debaromyces udenii* apresentam grande afinidade com a *Candida guillermondii* (o grau de semelhança das suas sequências é de aproximadamente 99,2%), formando um grupo filogenético distinto. *C. albicans* e quatro outras espécies de *Candida* foram agrupadas num cluster filogeneticamente próximo. Outro estudo incidindo na mesma zona codificante do DNA concluiu, através de análise comparativa de sequências, que as espécies do género *Saccharomyces* se encontram dispersas em vários grupos filogenéticos, tendo mesmo maior proximidade com espécies de *Candida*, *Kluyveromyces*, *Torulaspora* ou

Zygosaccharomyces. No entanto, foi observado que quatro espécies do género *Saccharomyces* (*S. cerevisiae*, *S. paradoxus*, *S. bayanus*, *S. pastorianus*) formam um grupo evolutivo muito homogéneo, em que as sequências dos seus gene 18S apresentam um grau de semelhança superior a 99,9% [32].

A identificação e estabelecimento de filogenias entre espécies de leveduras podem também ser realizados recorrendo ao gene codificador da cadeia 28S (26S em algumas espécies). Foi realizado um estudo que permitiu distinguir evolutivamente cerca de 500 espécies de leveduras apenas investigando uma fracção da sequência deste gene (cerca de 600 nucleótidos). Para algumas espécies, foram incluídas na análise sequências de diferentes estirpes. Ficou demonstrado que 55 espécies actualmente aceites não são mais do que sinónimos de outras espécies anteriormente descritas, dada a pequena diferença de nucleótidos observada. É este o caso da *Candida aaseri* e *Candida butyri* (zero nucleótidos de diferença); da *Candida humilis* e *Candida milleri* (um nucleótido diferente); ou da *Pichia subpelliculosa* e *Hansenula arabitolgenes* (nenhum nucleótido diferente) [33].

Estudos filogenéticos que comparem as regiões ITS1 e ITS2 entre os genes do rRNA permitem distinguir mais facilmente organismos muito semelhantes, dado que são sequências que apresentam maior variabilidade interespecífica do que os genes 18S e 28S. Deste modo, foi possível estabelecer as relações evolutivas e distinguir entre espécies semelhantes pertencentes ao géneros *Saccharomyces* [34], *Zygosaccharomyces* e *Torulaspora* [35]. No primeiro destes dois estudos, foi mais uma vez provada a grande proximidade filogenética entre as espécies *S. cerevisiae*, *S. bayanus*, *S. paradoxus* e *S. pastorianus* através da elevada semelhança (superior a 85% no ITS1 e a 95% no ITS2) entre as suas sequências. Por outro lado, as restantes *Saccharomyces* estudadas apresentam maiores diferenças na constituição nucleotídica destas regiões do seu genoma, com um grau de semelhança, entre qualquer par de espécies, sempre inferior a 50% no ITS1 e a 62% no ITS2.

Sempre que possível, é aconselhável utilizar como meio de comparação as sequências quer dos genes 18S e 28S, mais conservadas, quer das regiões ITS1 e ITS2, hipervariáveis. Os primeiros têm maior aplicação na dedução de relações filogenéticas entre espécies distantes enquanto que as segundas permitem uma maior distinção entre espécies muito próximas [35]. Num estudo que incidiu sobre 75 espécies de leveduras relacionadas com o género *Saccharomyces*, foram analisadas as sequências codificantes de

18S, 28S, ITS1, ITS2 bem como de outros genes nucleares e do genoma mitocondrial. As espécies foram divididas em 14 grupos evolutivos na árvore filogenética final. O que apresenta maior uniformidade é aquele ao qual pertencem as quatro espécies de *Saccharomyces* referidas anteriormente [32, 34] mais as espécies *S. mikatae*, *S. cariocanus*, *S. Kudriavzevii*. Estas sete espécies são designadas por *Saccharomyces sensu stricto* e estão filogenicamente separadas das outras (as *Saccharomyces senso lato*) que se encontram muito dispersas. Foi mais uma vez deduzido que o género *Kluyveromyces* apresenta grande heterogeneidade filogenética, tal como o género *Zygosaccharomyces*, embora este em muito menor grau. Portanto, este estudo vem acentuar a problemática da classificação actual das espécies agrupadas no designado "complexo de *Saccharomyces*" [36].

Análises mais vastas são construídas com base nestas sequências. Um estudo aproveitou todas as sequências completas de rRNA 18S de eucariótas sequenciadas até à data (2551) e construiu uma árvore filogenética de alto grau de complexidade, agrupando espécies em *clusters* [37]. Esta árvore representa as relações evolutivas entre todas as espécies analisadas e permitiu determinar quatro grandes grupos ou superfilos. Estes foram designados por: *Opisthokonta* (inclui os animais e fungos), *Plantae* (algas verdes, plantas terrestres e algas vermelhas), *Stramenopiles* e *Alveolata*. Embora tenha sido evidente que a filogenia baseada num único gene tem limitações na análise de uma amostra tão vasta, ficou igualmente provada a importância evolutiva dos genes do rRNA, nomeadamente do gene 18S, nos estudos de filogenia molecular. As sequências destes genes permitem fazer a distinção ao nível das espécies ou estirpes mas, adicionalmente, a sua análise pode ser utilizada na determinação de grupos filogenéticos de ordem superior.

Não é apenas em organismos eucariótas que as sequências dos genes do rRNA são investigadas em estudos filogenéticas. Estas análises são igualmente feitas para organismos procariótas com resultados que, novamente, vêm por em causa as classificações baseadas nas características morfológicas. Foi realizado um estudo sobre os gene 16S e 23S de bactérias pertencentes à ordem *Chlamydiales*. Com os resultados obtidos, foi possível propor alterações significativas à classificação taxonómica desta ordem, com a introdução de novas famílias, géneros e espécies ou a remodelação das existentes [38].

Dado o interesse despertado pela importância destes genes na filogenia molecular de espécies, foram construídos mapas de variabilidade sobre as sequências das cadeias de

rRNA 18S e 28S dos eucariótas e dos seus homólogos nos procariótas (16S e 23S). Estes mapas mostram a estrutura secundária adoptada pela molécula de rRNA de uma espécie (*S. cerevisiae* em eucariótas e *Escherichia coli* em procariótas). Cada nucleótido da sequência primária é simbolizado sob a forma de um ponto colorido, de acordo com o seu grau de conservação interespecífico deduzido por alinhamentos e algoritmos matemáticos [21, 39, 40]. Na 0, estão expostos os mapas de variabilidade das duas cadeias de rRNA, para eucariótas. Nos estudos subjacentes à construção dos mapas do rRNA 18S e 28S, foram comparadas as sequências de 500 e 77 espécies eucariótas, respectivamente. A visualização destas figuras permite inferir rapidamente as zonas de maior variabilidade, podendo funcionar como uma referência preliminar nos estudos de filogenia que envolvam estas sequências.

A grande proximidade filogenética das espécies de *Saccharomyces sensu stricto* [32, 34, 36] fazem deste grupo um importante indicador da validade das metodologias filogenéticos. Uma técnica molecular que permita diferenciar as suas espécies pode ser considerada um ferramenta valiosa em qualquer estudo que envolva identificação de organismos. Partindo deste pressuposto, neste trabalho são incluídas para análise as espécies *S. bayanus*, *S. cerevisiae*, *S. mikatae* e *S. paradoxus* recentemente sequenciadas [41].

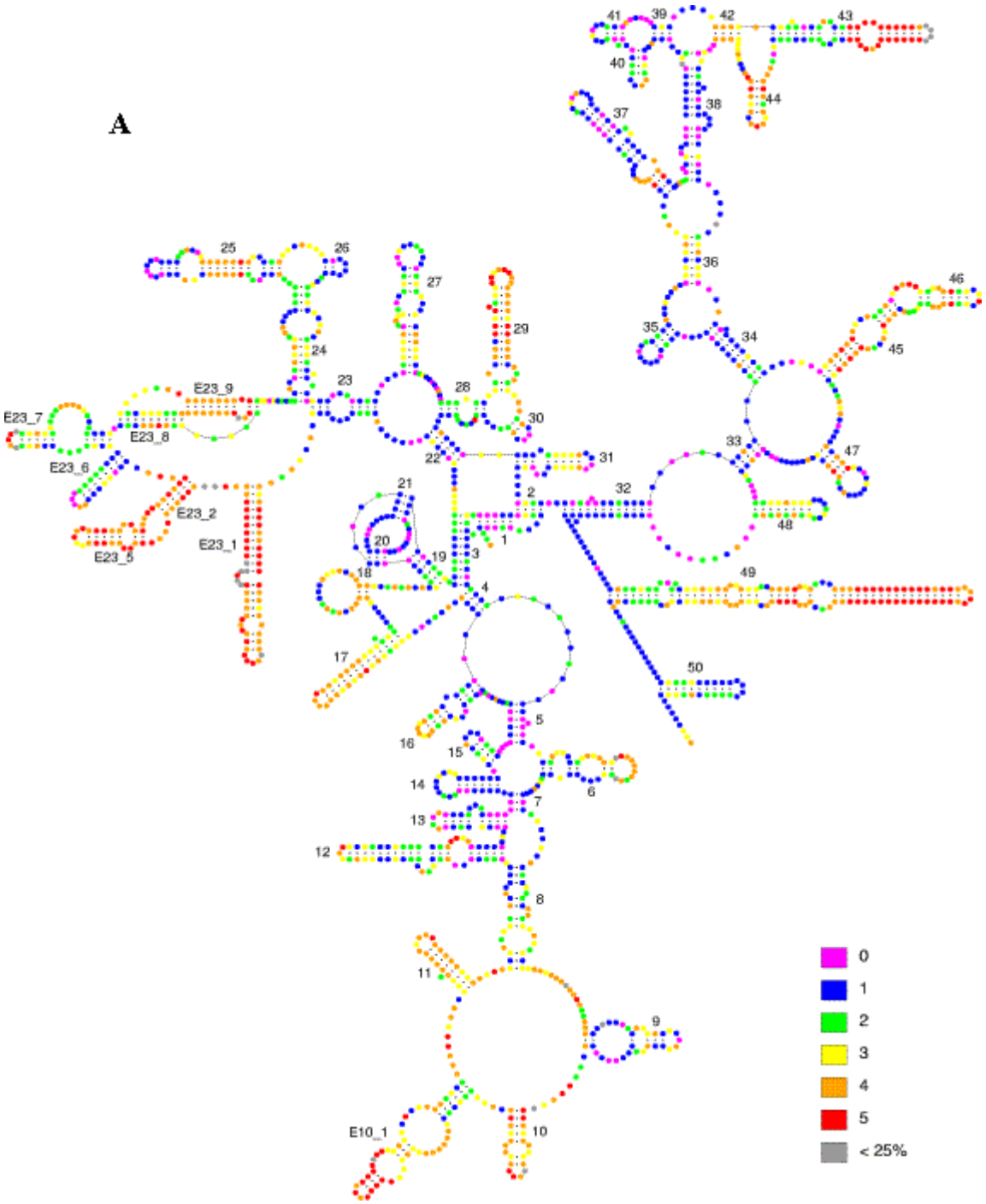

Figura 8 A – Mapa de variabilidade colorido sobreposto sobre a estrutura secundária do rRNA 18S de *S. cerevisiae*. Os nucleótidos, representados por pontos, foram subdivididos em cinco grupos de diferente variabilidade. As posições muito conservadas são indicadas a azul. Os pontos verdes, amarelos, laranjas e vermelhos representam posições com variabilidade crescente. As posições que apresentam conservação total entre espécies são indicadas a violeta. Os pontos cinzentos representam nucleótidos presentes em *S. cerevisiae* mas ausentes em mais de 75% das sequências de outras espécies. Adaptado de http://www.psb.ugent.be.

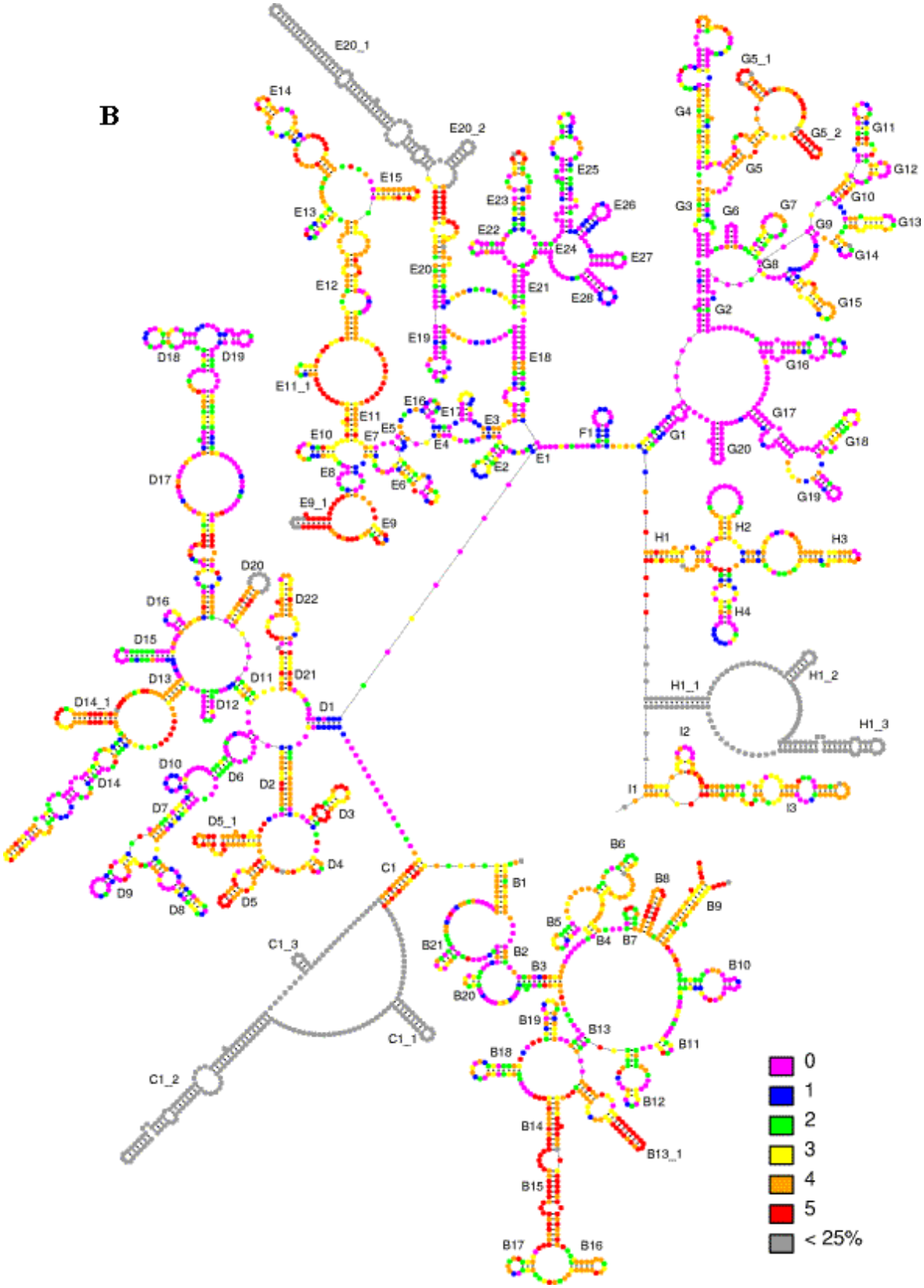

Figura 8 B – Mapa de variabilidade colorido sobreposto sobre a estrutura secundária do rRNA 28S de *S. cerevisiae*. Ver detalhes sobre cores na legenda anterior. Adaptado de http://www.psb.ugent.be.

C 3. Tecnologia de *Chips* de DNA e sua aplicação na expressão genética e diagnóstico molecular.

O crescimento vertiginoso do número de genomas sequenciados e a consequente identificação dos milhares de genes que codificam veio colocar um problema à área da genómica. Como realizar estudos sobre esta vasta amostra de sequências codificantes e não codificantes e deles extrair informação estruturada numa perspectiva global?

A tecnologia de *chips* ou *microarrays* de DNA desenvolvida durante os anos 90 do século passado veio proporcionar algumas soluções a esta problema. Através desta tecnologia, baseada na hibridação de ácidos nucleicos descrita anteriormente, é possível identificar sequências específicas na presença de milhares de sequências de uma amostra biológica complexa, numa única experiência. Esta possibilidade permite inúmeras aplicações nas áreas da biologia e da medicina.

3.1. Modo geral de funcionamento.

Os *chips* de DNA são pequenos suportes sólidos (1-5 centímetros quadrados de área) onde podem ser imobilizadas sequências de DNA de interesse designadas por sondas. O material de que são feitos os suportes é normalmente vidro, podendo também ser usado silício ou *nylon*. O vidro permite uma baixa fluorescência, transparência, resistência a altas temperaturas, rigidez física e uma variedade de modificações químicas possíveis na sua superfície. A natureza não porosa do vidro facilita o acesso às sondas nele imobilizadas, não ocorrendo difusão das soluções com os ácidos nucleicos [42-44].

Cada *chip* (Figura 9) pode ter entre algumas centenas até vários milhares de locais de teste. Estas pontos restringem-se a um tamanho de entre 10 a 500 micrómetros [45]. Cada ponto tem associado um conjunto de coordenadas precisas que o localizam no *chip* e apenas possui sondas de um determinado tipo. Estas sondas podem ser sequências de cDNA ou oligonucleótidos e encontram-se ligadas à superfície do *chip* por um dos terminais.

Figura 9 – *Chip* de DNA. São visíveis os inúmeros pontos de hibridação dispostos sobre a superfície do *chip*. Adaptado de http://www.dkfz-heidelberg.de.

Nos *microarrays* de cDNA, as sondas são obtidas a partir de clones de bactérias. Estes clones, com múltiplas cópias de um determinado gene, ficam assim disponíveis para serem utilizados em variadas experiências, incluindo o fabrico de *microarrays*. Determinadas instituições [46] possuem bancos de cDNA com clones catalogados correspondentes a todos os genes de um dado organismo (por exemplo, homem, ratinho, cão). Para se obterem as sondas a partir destes clones, o cDNA é extraído das células bacterianas, purificado e amplificado por PCR. As sondas são de seguida imobilizadas no suporte sólido através de *spotting* robotizado. O tamanho das sondas pode ser de algumas centenas de nucleótidos ou atingir os milhares de comprimento.

O utilização de oligonucleótidos no fabrico de *microarrays* tem vindo a aumentar, em parte devido à maior facilidade de manuseamento, sem as dificuldades logísticas associadas à manipulação de culturas de células. O seu emprego como sondas requer apenas o conhecimento prévio da sequência de nucleótidos necessária à hibridação. Deste modo, o desenho de oligonucleótidos tira partido do contínuo enriquecimento das bases de dados de sequências e mapas genómicos. As sondas deste tipo são mais pequenas (normalmente entre 15 e 80 - mer) e são sintetizadas *in situ* directamente no suporte do *chip* ou pelos métodos convencionais e de seguida imobilizadas no suporte sólido [47].

A síntese directa dos oligonucleótidos sobre o *chip* pode ser feita pelo método de fotolitografia. Este processo utiliza de uma superfície de vidro coberta com uma camada fotossensível. Em determinados pontos faz-se incidir uma radiação que derrete esta camada. Posteriormente, adiciona-se uma solução com um tipo de nucleótido (A, G, T ou C) que se liga nesses pontos à matriz do suporte. Após esta fase, a substância não ligada é lavada. De seguida, este ciclo é repetido para outras posições do *chip*, até se ter iniciado a síntese de todos os seus oligonucleótidos. Inicia-se então a deposição dos segundos nucleótidos sobre os primeiros, segundo o mesmo princípio: irradiação localizada, ligação, lavagem. De referir que todos os nucleótidos adicionados estão modificados para reagirem às radiações incidentes da mesma forma que a camada fotossensível do suporte. A síntese continua, com a adição de novas camadas de nucleótidos até se constituírem as sondas oligonucleotídicas completas [48]. Este método permite a construção de um *chip* com alta densidade de pontos e com múltiplas sondas de sequências diferentes [49].

A imobilização de oligonucleótidos pré-sintetizados ou cDNA no *chip* pode ser feito por impressão directa das sondas usando agulhas ou pinos que transferem as sondas de uma placa com 96 ou 384 poços para o suporte sólido. O diâmetro e forma do pino, a viscosidade da solução e as características do suporte determinam o volume transferido e o tamanho da dispersão da solução. Outra tecnologia semelhante utiliza uma espécie de jacto para pulverizar as sondas de oligonucleótidos no *chip* [47].

Estes processos complexos de síntese e/ou imobilização das sondas no suporte do *chip* são executados a alta velocidade por robôs mecanizados, altamente precisos e controlados por computador. Com o evoluir das tecnologias mecânicas, electrónicas e informáticas os *microarrays* tenderão a ficar mais compactos, com pontos de menor diâmetro e em maior número por unidade de área.

A utilização de cDNA no fabrico de *chips* tem como vantagem não ser necessário o conhecimento prévio da sequência total. É possível realizar a sua amplificação por PCR recorrendo a *primers* universais e a sequenciação de genes que se revelem importantes pode ser realizada após a análise do *microarray*. A extensão elevada das sondas de cDNA permite uma maior especificidade de hibridação com a amostra, evitando a hibridação cruzada, ou seja, a ligação a sequências não totalmente complementares. Uma das principais desvantagens são os anteriormente referidos protocolos laboratoriais de extracção, purificação e amplificação de cada tipo de cDNA, especialmente se o número de

sondas exigido no *chip* for muito elevado. A presença recorrente de contaminações nos clones de cDNA (por exemplo, plasmídeos incorrectos) pode igualmente prejudicar a utilidade da sua utilização em *microarrays* [43, 50, 51].

Nos *chips* com sondas de oligonucleótidos, a menor extensão destes pode ser um óbice na precisão da hibridação, com ocorrência significativa de hibridação cruzada. Esta desvantagem, porém, pode ser total ou parcialmente anulada com a utilização de vários (tipos de) oligonucleótidos direccionados à mesma sequência da amostra. Esta estratégia permite também um maior conhecimento da sequência alvo hibridada, como seja a distinção entre diferentes mRNA's resultantes de *alternative splicing*. A possibilidade de desenhar oligonucleótidos do mesmo tamanho e com aproximadamente a mesma Tm e conteúdo em G+C permite que todas as reacções de hibridação no *chip* se realizem sob as mesmas condições. A deposição de uma dada concentração de oligonucleótidos no suporte de um *microarray* é igualmente tida como mais fácil em comparação com o mesmo processo nos *chips* de cDNA. Os oligonucleótidos são de cadeia única, não havendo necessidade de adicionar etapas de desnaturação, nem ter em atenção a possibilidade de renaturação, tão características dos ácido nucleicos de cadeia dupla, como o cDNA. Um *microarray* de oligonucleótidos permite também uma maior densidade de sondas imobilizadas sobre o suporte em relação a um *microarray* de cDNA, com os decorrentes benefícios de miniaturização e alargamento do espectro de sequências em análise. Finalmente a síntese de oligonucleótidos, *in situ* no *chip* ou pelos métodos convencionais é um processo simples e relativamente barato que permite a adaptação e modulação da plataforma de *microarrays* a uma grande variedade de experiências [43, 50, 51].

Para além da construção do *chip*, é necessário ter em conta a importância da preparação da amostra biológica que será testada. As suas células podem ter origens muito variadas, incluindo linhas celulares de laboratório, tecidos de pacientes, fragmentos de plantas ou amostras ambientais. Frequentemente, é necessário recorrer a processos de purificação para libertar a amostra de vários tipos de impurezas que deturpem os resultados da experiência. Se a concentração dos ácidos nucleicos (DNA ou RNA) for baixa, é igualmente necessário proceder à sua amplificação por PCR. De seguida, realiza-se a marcação das cadeias com corantes fluorescentes específico tais como o Cy3 ou Cy5 [52].

Posteriormente, se a amostra for incubada no *chip* é esperado que as suas sequências hibridem com as sondas presentes nos pontos que tenham uma sequência complementar de bases. Após ser removido todo o DNA ou RNA da amostra que não ligou ao *chip*, é possível detectar a ocorrência de hibridação em cada ponto, através de um *scanner* de laser. Com a passagem do(s) laser(s) sobre a superfície do *chip*, os pontos que emitem fluorescência são aqueles aos quais se ligaram cadeias marcadas da amostra.. As intensidades das radiações emitidas em cada ponto são estimadas automaticamente por meio de *software* de análise e os seus valores são armazenados conjuntamente com as coordenadas dos pontos. Estes resultados podem depois ser integrados revelando a informação pretendida [47].

3.2. Tipos e aplicações de *chips*.

Uma aplicação muito comum dos *chip* de DNA é o estudo dos níveis de expressão de genes. Uma experiência típica está representada na Figura 10. São consideradas dois grupos de células com a mesma origem: do tipo 1, células em condições controlo e do tipo 2, células sob uma determinada condição especial. O objectivo da experiência é determinar a taxa de expressão de um conjunto de genes, ou seja, verificar como é que as células alternam a transcrição de genes em resposta a modificações ambientais.

O mRNA das células é isolado e é utilizado como molde para gerar cDNA que será marcado com um fluoróforo. É normalmente utilizado o cDNA por ser mais estável do que as moléculas de mRNA. As células do tipo 1 vêem o seu cDNA marcado com a cor verde e as do tipo 2 com um marcador vermelho. As duas soluções são misturadas e incubadas com um *microarray* que contém imobilizados os genes em pontos diferentes. Algumas das moléculas marcadas de cDNA hibridam então com as sondas suas complementares. Após esta fase, os constituintes da mistura que não se encontrem ligados ao *chip* são removidos por lavagem.

De seguida, uma imagem de fluorescência do *chip* é criada por um *scanner* sob a irradiação sequencial de um laser vermelho e de outro verde. Para cada ponto, a razão entre as duas cores é calculada e o ruído de fundo subtraído. São criadas tabelas com estes valores que revelam o tipo de cDNA que hibridou em cada ponto. Estes valores têm uma relação directa com os níveis de expressão de cada gene nas duas condições. Se um ponto se apresentar verde, significa que ocorreu hibridação do cDNA das células do tipo 1,

indicando que o gene apenas é expresso na condição normal. O contrário é verificado quando a cor detectada é vermelha. Quando num ponto é observada uma tonalidade amarela (combinação verde mais vermelho), então, houve hibridação com os dois tipos de cDNA porque o gene correspondente é expresso tanto na condição normal como na condição especial. Em determinados casos, ligeiras diferenças na tonalidade observada podem representar pequenas mas importantes variações na expressão de um gene. Se, por outro lado, o ponto se apresenta negro, sem cor, isso significa que não ocorreu hibridação de nenhum cDNA complementar, logo não terá ocorrido expressão do gene em nenhuma das duas condições.

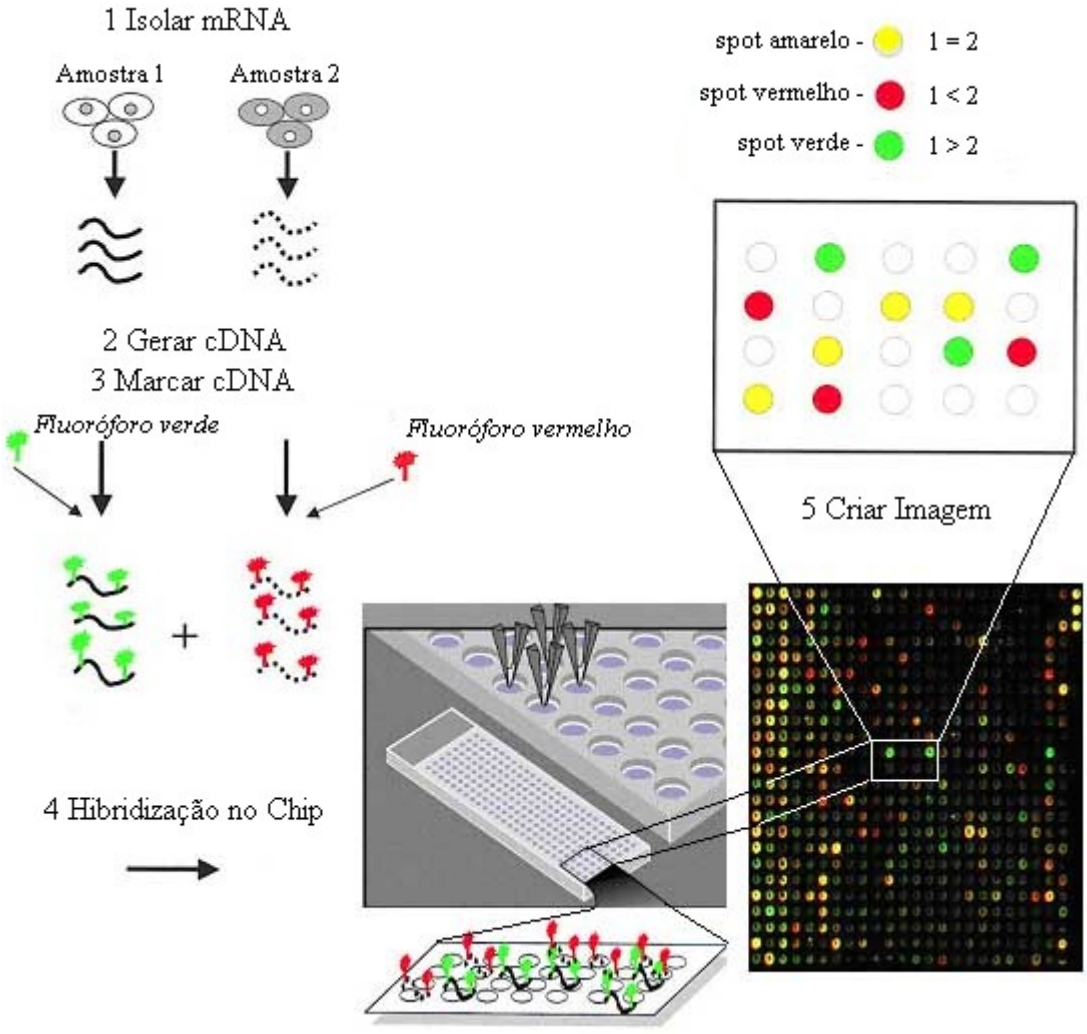

Figura 10 – Aplicação de um *Chip* de DNA na quantificação da expressão de genes (ver detalhes no texto). Adaptado de http://www.fao.org.

O exemplo anterior é apenas uma demonstração básica de um *chip* de análise de expressão de genes. No entanto, o seu princípio básico, a quantificação do mRNA sintetizado, pode ser aplicado a experiências com um grau de complexidade muito mais elevado. Actualmente, é possível desenhar, construir e analisar com sucesso *chips* que detectem a expressão de todos os genes de um genoma [53]. *Chips* com sondas representativas dos cerca de 6000 genes de *S. cerevisiae* foram usados para determinar as alterações na expressão genética que ocorrem quando são sujeitas a mudanças bruscas de temperatura [54]; quando as leveduras transitam do estado de fermentação de glucose para a respiração [55]; ou quando alternam entre as fases do seu ciclo celular [56]. Outros estudos recorrem a *chips* para monitorizar os níveis de expressão dos genes de células pertencentes a tecidos humanos doentes. Estas análises são realizadas com particular incidência no estudo de cancros, patologias que apresentam frequentemente uma origem genética multifactorial [57-60].

Outra importante aplicação dos *chips* de DNA é no desenvolvimento de novos medicamentos. Esta tecnologia pode ser aplicada no rastreio do genoma completo de células doentes e, através do perfil de expressão, permitir seleccionar os genes intervenientes na patologia [61]. Em fases mais adiantadas, *chips* de DNA podem ser utilizados para determinar os efeitos, positivos ou adversos, que compostos têm sobre a condição geral do organismo doente [62, 63]. Os *chips* também podem ser aplicados na identificação de posições de um genoma que apresentem variações polimórficas de um único nucleótido entre indivíduos da mesma espécie (*Single-Nucleotide Polymorphisms*, SNP) [64]. Estas pequenas variações são, em muitos casos, o motivo por que certas drogas actuam eficazmente em determinadas pessoas e não noutras [65]. A um nível mais básico, a alteração de um nucleótido por outro é, em determinadas condições, a causa de algumas doenças.

Os *chips* de DNA foram recentemente introduzidos no estudo de organismos patogénicos e da sua interacção com os hospedeiros, particularmente o homem. Mais uma vez, a grande vantagem desta tecnologia em relação aos métodos clínicos tradicionais é a sua capacidade de detectar facilmente a presença de milhares de ácidos nucleicos em simultâneo.

Com recurso a *chips* de DNA é possível traçar o perfil de expressão dos genes de um organismo patogénico. Quando um microrganismo invade o hospedeiro, sofre um processo de adaptação ao novo nicho ecológico. Este processo envolve o aumento da expressão dos genes essenciais e a diminuição daqueles que deixaram de ser necessários. O mesmo princípio é aplicado aos genes de virulência que são sujeitos a mecanismos de regulação que garantem uma expressão adequada às condições ambientais do hospedeiro. Ao ser registado o perfil de expressão de todos os genes de um organismo patogénico, é possível determinar aqueles que são frequentemente activados durante os processos de infecção [66-68]. Uma característica dos genes de virulência de várias espécies é a sua co-regulação. Este pressuposto permite deduzir, em determinados circunstâncias, que um determinado gene é importante na infecção quando os seus níveis de expressão acompanham os de outro gene com virulência conhecida [69]. Vários estudos têm sido realizados sobre os mecanismos genéticos de adaptação de organismos patogénicos ao ambiente interno de um hospedeiro. Entre outros, foram analisados os níveis de expressão da levedura *Candida albicans* em condições semelhantes às do sangue humano [70]; as alterações, ao longo do tempo, do perfil de expressão dos genes da bactéria *Mycobacterium tuberculosis* (responsável pela tuberculose) numa infecção em ratinhos [71]; e a expressão de todos os genes do citomegalovírus humano [72].

O estudo dos níveis de expressão dos microrganismos patogénicos através de *chips* de DNA pode revolucionar o desenvolvimento de novos medicamentos para o combate a infecções. Se uma espécie infecciosa for cultivada na presença de um composto químico, as alterações dos níveis de expressão de todos os seus genes podem ser monitorizadas num *chip*. A constituição desta "assinatura" específica para uma droga permite aprofundar o conhecimento do seu efeito nocivo nas vias metabólicas do microrganismo, facilitando a validação ou rejeição do seu uso como medicamento [69]. Estudos deste tipo tem vindo a ser realizados sobre várias espécies, como o *Mycobacterium tuberculosis* [73, 74], *Candida albicans* [75], *Saccharomyces cerevisiae* [76] ou *Streptococcus pneumoniae* [77]. A possibilidade, anteriormente referida, de "filtrar" todo o genoma destes microrganismos em busca dos genes activados durante o processo de infecção permite, igualmente, a selecção desses genes de virulência como candidatos a alvos no desenvolvimento de drogas [69].

A monitorização dos níveis de expressão através de *chips* de DNA também é realizada nas células do hospedeiro [78]. O modelo básico consiste em medir a expressão

de todos os genes destas células antes e depois de serem infectadas por um microrganismo [69]. Com esta análise, é possível identificar os genes cuja expressão aumenta ou diminui, durante o processo de infecção. Estas observações permitem identificar os genes importantes nos mecanismos de defesa do hospedeiro e fornecem informação indirecta sobre as suas funções específicas [66-68]. Ao mesmo tempo, a partir do conhecimento da resposta do hospedeiro é possível aprofundar a percepção dos mecanismos de patogenicidade do invasor [69] e até mesmo avançar com um prognóstico – vírus, bactérias ou fungos patogénicos originam diferentes "assinaturas" nos níveis de expressão do hospedeiro. Os estudos da resposta do hospedeiro humano incidem frequentemente, como seria de esperar, nas células envolvidas na imunidade, como é o caso dos macrófagos [79]. A exposição de células de hospedeiro a bactérias [80] ou a vírus [81, 82] é estudada e os efeitos a nível da expressão genética são determinados usando *chips* de DNA.

3.2.1. *Chips* de diagnóstico.

O *chips* de diagnóstico, são um caso particular desta tecnologia de hibridação de ácidos nucleicos porque, ao contrário dos estudos da expressão de genes, não há necessidade de se recorrer a duas amostras marcadas (condição e controlo) para a obtenção dos resultados, mas apenas a uma. Esta variante permite determinar a presença de uma dada espécie microbiana ou de um vírus numa amostra, unicamente pela detecção de sondas específicas para cada espécie ou tipo de vírus. Deste modo, a análise do *chip* é realizada num modo binário, tendo em conta a presença ou ausência de sinal nos pontos identificativos de uma espécie. A presença de sinal indica que ocorreu hibridação entre a sonda e a amostra marcada, logo a espécie encontra-se presente (Figura 11).

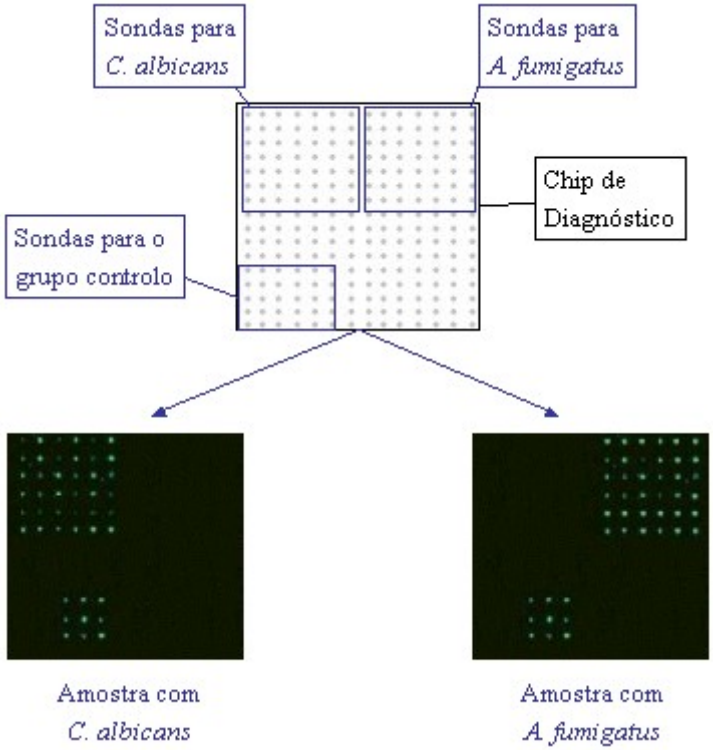

Figura 11 – Exemplo esquemático do funcionamento de um *chip* de diagnóstico capaz de identificar as espécies *Candida albicans* e *Aspergillus fumigatus* em amostras clínicas. O *chip* é desenhado com um conjunto de sondas específicas para cada espécie e um conjunto de sondas de controlo. Após a incubação do *chip* com uma amostra, é possível verificar quais as espécies contaminantes nela presentes através do sinal dos pontos de hibridação correspondentes.

A selecção do tipo de sondas a colocar em cada ponto de hibridação é um dos factores fundamentais na capacidade de diagnóstico de um *chip* de DNA. Na maioria dos estudos, as sequências escolhidas têm necessariamente que ser específicas, ou seja, a ordem exacta de nucleótidos apenas pode ser encontrada no genoma de uma única espécie. Este factor pode ser considerado em absoluto (para todas as espécies existentes e com sequências nucleotídicas conhecidas) ou de forma relativa (considerando apenas as espécies em estudo, ou um grupo taxonómico particular). Se a sonda não for específica, pode ocorrer hibridação cruzada com sequências de outras espécies levando à observação de falsos positivos. Outros factores condicionantes são igualmente tidos em conta no desenho das sondas, como a sua extensão e composição, a temperatura de fusão de cada sequência ou a possibilidade de formação de estrutura secundária. Estes e outros aspectos do desenho de *chips* de diagnóstico serão aprofundados e discutidos posteriormente.

A aplicação de *chips* de DNA na identificação e genotipagem de microrganismos é uma área de investigação que tem sofrido grandes avanços nos últimos anos. É um método

rápido e fiável de diagnóstico, em amostras clínicas, ambientais, alimentares ou outras. Vários estudos tem sido realizados com o intuito de aperfeiçoar a tecnologia e os protocolos laboratoriais, a nível do número de espécies abrangidas, do grau de especificidade e sensibilidade da hibridação e da objectividade, fiabilidade e reproducibilidade dos resultados.

Nas metodologias de diagnóstico são preferencialmente utilizados os *chips* de oligonucleótidos porque são capazes de distinguir pequenas diferenças, como mutações de um único nucleótido, entre sequências idênticas, ao contrário dos *chips* de cDNA [68]. Esta vantagem das sondas de oligonucleótidos é essencial na identificação diferenciada de espécies semelhantes ou na distinção entre estirpes ou genótipos da mesma espécie (genotipagem).

Um grupo taxonómico sobre o qual tem incidido um grande número de análises é o dos vírus. Foi desenvolvido um método baseado em *chips* que faz a detecção e genotipagem dos rotavírus humanos do grupo A. Estes vírus com genoma de RNA apresentam uma grande taxa de mutações, logo um alto nível de polimorfismo entre si. No entanto, com este método foi possível definir o genótipo correcto das 40 estirpes em análise [83]. Outro estudo permitiu a identificação inequívoca de 20 amostras contendo vários genótipos do mesmo vírus. Os resultados foram corroborados pelos métodos tradicionais de genotipagem [84]. Wang e colaboradores (2002) desenvolveram uma metodologia capaz de identificar, num único *microarray* de oligonucleótidos, vírus muito diversos. Para testar o modelo, foi desenhado um *chip* com sondas para centenas de vírus, incluindo praticamente a totalidade dos vírus do tracto respiratório humano como os rinovírus e enterovírus. A incubação de várias amostras contaminadas com vírus de identidade conhecida permitiu observar a grande exactidão do *chip* desenhado e levou a concluir que qualquer vírus com genoma conhecido pode ser detectado por esta aproximação [85]. Outro estudo com *chips* de DNA incidiu sobre vírus do tipo *influenza*. Este vírus apresenta grande diversidade a nível dos hospedeiros infectados e da sua sequência genética. Os *chips* desenhados mostraram-se capazes de distinguir, de modo preciso, as várias espécies e subtipos (de hemaglutinina) do vírus *influenza* [86].

Da mesma forma que os vírus, também a identificação de bactérias em amostras clínicas tem sido alvo de inúmeros estudos baseados em *chips* de DNA com sondas oligonucleotídicas. Um estudo teórico incidiu na identificação de estirpes de *Escherichia*

coli e outras bactérias entéricas patogénicas em amostras de identidade anteriormente conhecida [87]. Outra análise permitiu a identificação com sucesso de *Escherichia coli*, três espécies de *Shigella* e sete estirpes de *Salmonella enterica* [88]. Foi igualmente desenvolvido um método baseado num *chip* de oligonucleótidos para a detecção de 20 espécies bacterianas da microflora intestinal humana. Foram recolhidas amostras fecais de indivíduos saudáveis, aos quais foram detectadas populações equilibradas das várias bactérias, e de um indivíduo com diarreia crónica, ao qual foi detectada uma microflora anormal, com predominância de uma única espécie sobre as outras [89]. A tecnologia de *chips* de DNA também se revelou útil na rápida detecção e distinção de seis espécies semelhantes de *Listeria* em culturas laboratoriais [90]. Outro estudo demonstrou a importância da aplicação de *microarrays* no diagnóstico, através da identificação diferenciada das bactérias termofílicas *Campylobacter jejuni*, *C. coli*, *C. lari* e *C. upsaliensis* de elevada incidência epidemiológica [91]. Um *array* de oligonucleótidos específicos imobilizados sobre uma membrana de *nylon* foi utilizado para identificar várias espécies de bactérias em culturas de sangue, tendo os resultados sido satisfatórios, com a identificação incorrecta de apenas 8 em 158 amostras [92].

A identificação de microrganismos em amostras ambientais com recurso a *chips* de DNA também assume grande importância, devido à fiabilidade e rapidez desta tecnologia. Análises ao conteúdo bacteriano de alimentos vegetais [93], amostras de ar [94], solo [95] e sedimentos de aquíferos [96-99] que incluíam metanotrófos [95], procariótas redutores de sulfato [96] e populações microbianas mais complexas, foram estudadas com sucesso usando *chips* de DNA.

A metodologia empregue nas análises ecológicas assemelha-se em vasta medida à utilizada sobre amostras clínicas. Em ambos os casos, faz-se a preparação da amostra de DNA, purificação e marcação do DNA e em paralelo desenha-se e constrói-se o *chip*. Os dois componentes são de seguida incubados, ocorrendo hibridação entre ácidos nucleicos complementares. Uma imagem do *chip* é obtida usando um *scanner* laser e os resultados são analisados com o menor grau possível de subjectividade. Caso seja necessário, recorre-se ao refinamento da técnica, como, por exemplo, a selecção de sondas mais específicas para melhorar a resolução da experiência.

A escolha das genes a utilizar como marcadores específicos é crucial no desempenho dos *chips* de diagnóstico. Os genes do rRNA, em particular a cadeia da SSU,

têm sido consistentemente utilizados em muitos destes estudos [89, 92-94, 96, 97], devido à sua elevada conservação e presença de regiões variáveis entre espécies (conforme foi descrito na secção 2.5). Outra vantagem destes genes é a grande quantidade de sequências disponíveis nas bases de dados [100, 101]. No entanto, em determinados casos, a diferenciação entre subespécies da mesma espécie ou entre espécies muito semelhantes não é conseguida através deste gene. Nos organismos eucariótas, uma alternativa será a utilização das regiões ITS1 e ITS2, que apresentam um grau mais elevado de variabilidade, ou a conjugação de todos os genes do rRNA (SSU, ITS1, 5.8S, ITS2 e LSU).

Outra estratégia utilizada em alguns estudos foi a utilização de genes funcionais como marcadores da identificação de espécies. Estes incluem o gene codificador da monooxigenase de metano (pmoA) [95], os genes codificadores de enzimas envolvidas no ciclo do azoto (incluindo reductase de nitrito, monooxigenase de amónia ou nitrogenease) [99] e o gene gyrB (codificador de uma subunidade proteica da enzima "DNA *gyrase*") [88]. Estes estudos permitem limitar o estudo ao grupo de espécies que contêm esses genes. Outros estudos incidiram sobre genes relacionados com a virulência e codificadores de determinantes antigénicos dos organismos patogénicos a identificar [87, 90]. Esta aproximação, permite ao mesmo a identificação das espécies e dos seus factores de virulência.

A extensão da sequência das sondas é uma característica que influencia em grande medida a especificidade e a sensibilidade da hibridação. Oligonucleótidos pequenos (menos de 25 nucleótidos) tem mais capacidade de fazer a distinção entre duas sequências alvo semelhantes do que oligos grandes (superior a 50 nucleótidos). Por outro lado, estes são mais sensíveis e permitem a detecção de baixas concentrações de sequências complementares numa mistura complexa de ácidos nucleicos. Outros parâmetros da selecção de sondas para um *chip* de diagnóstico serão debatidos mais à frente na discussão dos resultados.

Um problema comum nos estudos clínicos e ambientais de diagnóstico é a baixa densidade de células do microrganismo infeccioso nas amostras. Isto implica que o DNA é extraído em pequenas quantidades. A incubação directa desse extracto com o *chip* iria originar níveis mínimos de hibridação, logo sinais fluorescentes muito ténues nos pontos de hibridação. Outro inconveniente é a presença inevitável de grandes quantidades de

DNA do hospedeiro (amostras clínicas) ou de outros microrganismos (amostras ambientais) que iriam introduzir ruído na hibridação, por perturbações espaciais ou hibridação cruzada. Por este motivo, um passo comum na maioria destes estudos envolve a amplificação, por *multiplex*-PCR, dos genes marcadores das espécies a identificar. Nesta etapa, sintetiza-se DNA marcado com fluoróforos a partir do DNA [87-96, 98, 99] (ou cDNA a partir do RNA, por RT-PCR [83-86]) das espécies presentes. A selecção do(s) par(es) de *primers* para a amplificação é, tal como a das sondas, um procedimento que requer extremo cuidado. Pretende-se amplificar unicamente os genes das espécies em estudo e com o número mínimo possível de pares de *primers* para evitar formação de produtos de PCR indesejáveis [102] (tema mais aprofundado na secção de resultados e discussão).

3.3. Análise de resultados.

Para obter informação fiável a partir de um *chip* de DNA é necessário proceder a uma análise cuidadosa dos seus resultados. Isto implica fazer o registo rigoroso dos sinais de fluorescência de cada um dos seus pontos de hibridação e atribuir um significado biológico aos valores e padrões obtidos.

O processo de análise é levado à prática com o auxílio de *software* de análise, frequentemente fornecido pelos fabricantes de *scanners* ou distribuído por projectos independentes de *freeware*. Alguns exemplos de programas disponíveis são o GeneSpring da *Silicon Genetics* (http://www.silicongenetics.com) e o GeneCluster do *Whitehead Institute* (http://www.genome.wi.mit.edu/cancer/software/software.html). É igualmente possível utilizar software universal de estatística na análise de resultados *microarrays*, como o Matlab (http://www.mathworks.com) ou o pacote R (http://www.r-project.org).

O processamento da imagem do *chip*, inclui a normalização dos sinais registados. A normalização ajusta as diferenças na eficiência da marcação e detecção e na quantidade de ácidos nucleicos presentes na amostra [103]. Na análise de *chips* de expressão este ajuste assume uma particular importância pois estão em causa duas condições laboratoriais diferentes e é essencial que possam ser comparadas directamente.

Nos *chips* de expressão, os resultados correspondentes a cada gene são representados como uma "razão", que não é mais do que o valor normalizado do nível de expressão de um gene na condição em estudo dividido pelo seu valor normalizado no

controlo. Esta proporção permite identificar os genes que na condição em estudo viram os seus níveis de expressão elevados ou diminuídos e a grandeza dessa variação [103]. A partir desta fase e dependendo dos objectivos, existe uma grande variedade de estratégias e metodologias matemáticas para extrair informação com potencial valor biológico de uma experiência com *chips* de DNA. Alguns dos possíveis objectivos são identificar genes que sobressaiam para estudos futuros; comparar níveis de expressão entre genes diferentes, identificar genes que pertençam a vias metabólicas; definir conjuntos de genes com comportamento semelhante.

Uma das metodologias matemáticas mais úteis na análise de *chips* de DNA é o *clustering* porque permite identificar padrões nos resultados de expressão de genes. A maioria das análises baseadas nestes algoritmos são hierárquicas, com um número de classes relacionadas entre si de modo semelhante a uma classificação filogenética. Os genes pertencentes a uma classe possuem algum grau de semelhança [104]. Outras aproximações de análise possíveis são *Class Prediction* e *Self-Organizing Maps* [105]. Nestas e noutras metodologias matemáticas de análise, um dos principais desafios é aplicar os algoritmos de forma apropriada para que os resultados façam sentido a nível biológico [103].

Nos *chips* de diagnóstico, a análise é relativamente mais simples do que nos *chips* de expressão pois, como foi referido, é determinado unicamente se um ponto de hibridação exibe ou não fluorescência. No entanto, a intensidade do sinal emitido por cada ponto positivo (onde ocorreu hibridação específica) pode ser variável dentro do mesmo *chip*, por motivos diversos que incluem a diferente especificidade e sensibilidade de cada tipo de sonda e a concentração dos diversos fragmentos de ácido nucleico da amostra. Por outro lado, os pontos "negativos" também podem apresentar fluorescência resultante de hibridação cruzada. Por estes motivos, é necessária a inclusão de controlos positivos e negativos no *chip* que permitam identificar o limiar de fluorescência a partir do qual se considera que ocorreu hibridação específica.

3.4. Normas e Bases de Dados.

Um dos grandes desafios inerentes à utilização de *chips* de DNA num laboratório é decidir a melhor forma de armazenar os milhares de resultados produzidos em cada experiência. Para isso, é necessário escolher que parte dos dados guardar, para mais tarde

ser possível reproduzir ou re-analisar os resultados. Outro problema dos *chips* de DNA é a tendência de cada plataforma laboratorial produzir resultados difíceis de reproduzir noutras plataformas. Para colmatar estas questões protocolares, têm vindo a ser estabelecidos um conjunto de normas para descrever as experiências de *chips*, sistemas para tratamento e transferência de dados e repositórios públicos para o seu armazenamento e pesquisa [106, 107].

A sociedade MGED (*Microarray Gene Expression Data*) tem promovido uma série de projectos com grande apoio por parte da comunidade científica. Alguns dos mais significativos são o MIAME (*Minimum Information About a Microarray Experiment*) e o MAGE (*Microarray Gene Expression*).

O projecto MIAME define a informação acerca de uma experiência com *chips* que deve ser sempre incluída nas bases de dados e também fornece linhas de direcção aos autores de artigos científicos baseados nessas experiências [108]. A aplicação destas normas assegura, em princípio, que os dados possam ser facilmente interpretados, que os resultados possam ser independentemente verificados e que a informação esteja estruturada de forma a permitir a sua pesquisa e análise automática [108].

O MAGE tem como objectivo expor normas padrão para a representação de dados e foi desenvolvido sob duas perspectivas. O MAGE-OM (MAGE *Object Model*) é uma especificação, orientada a objectos, que pode servir de modelo às bases de dados de *microarrays*; exibe os dados associados a cada experiência e os relacionamentos entre cada tipo de dados. O MAGE-ML (*Microarray Gene Expression Markup Language*) é a implementação em *software*, pela tradução do MAGE-OM para a linguagem XML, que define o formato comum de transferência entre bases de dados [109].

Estas normas têm vindo a ser aplicadas em grande escala, a nível mundial, pelos grupos de investigação e também pelas instituições que gerem e mantêm as bases de dados de experiências de *microarrays*. Exemplos de repositórios públicos importantes que suportam as normas do MGED são o ArrayExpress instalado no *European Molecular Biology Laboratory – European Bioinformatics Institute* (EMBL-EBI) [110], o *Gene Expression Omnibus* (GEO) do *National Center for Biotechnology Information* (NCBI) [111] e o *Center for Information biology Gene Expression database* (CIBEX) pertencente ao *DNA Data Bank of Japan* (DDBJ) [112].

De salientar que, embora as normas do MGED para *microarrays* tenham sido criadas particularmente para experiências de expressão de genes, são adaptáveis a outras experiências, como os *chips* de diagnóstico. Por conseguinte, as bases de dados referidas permitem a submissão de dados resultantes dessas experiências [110].

Actualmente, é cada vez maior o número de publicações científicas que considera a submissão, dos dados relevantes de cada estudo, para um destes repositórios como uma requisito obrigatório para a validação dos artigos recebidos [113]. Por esse motivo e pelas vantagens científicas inerentes, os laboratórios devem tentar obedecer a estas normas na realização de qualquer experiência com *chips* de DNA.

D 4. A importância da Bioinformática na genómica e no diagnóstico molecular.

Avanços nas metodologias científicas nas área da biologia molecular e genómica têm conduzido a um aumento exponencial da informação biológica gerada pela comunidade científica mundial. Neste panorama, destacam-se os projectos de sequenciação e anotação dos genomas de várias espécies de organismos biológicos, sendo o do homem o que maior quantidade de informação gerou. Esta avalanche de informação genómica requer, por seu lado, novas tecnologias e metodologias para armazenar, organizar e analisar esses dados.

A Bioinformática representa uma nova área interdisciplinar que utiliza aproximações computadorizadas para responder a questões biológicas. Para tal, os investigadores tiram partido de conjuntos vastos e complexos de dados, de forma rigorosa com o objectivo de obter conclusões biológicas válidas [114].

No início, as preocupações da bioinformática referiam-se unicamente à construção e manutenção de bases de dados de armazenamento de informação biológica. Com o enriquecimento destas mesmas bases de dados, adveio uma nova necessidade: combinar a informação nelas presente. Portanto, actualmente a tarefa mais premente envolve a análise e interpretação de vários tipos de dados, incluindo sequências de nucleótidos ou resíduos de aminoácidos e estruturas de proteínas. O objectivo último desta área de investigação é criar uma perspectiva globalizada da qual se possam discernir princípios unificadores da biologia.

A informação biológica é analisada através do desenvolvimento de novos algoritmos matemáticos e metodologias estatísticas com os quais se determina a existência de relações entre membros de conjuntos de dados diferentes. Estes processos podem ter uma miríade de finalidades, incluindo o alinhamento de sequências, a análise total ou parcial de um genoma, a localização de um gene numa sequência, a previsão da estrutura ou função de uma proteína, o agrupamento de genes ou proteínas em famílias e a determinação filogenética de um organismo.

4.1. Bases de dados biológicas.

A produção maciça de informação biológica levanta vários desafios, nomeadamente armazenar, catalogar e disponibilizar de forma eficiente essa informação.

Para tal, têm vindo a ser desenvolvidas e implementadas bases de dados públicas e, na grande maioria dos casos, de livre acesso. Estes repositórios permitem, regra geral, a sua consulta através da Internet pelo protocolo de transferência *World Wide Web* e em alguns casos adicionalmente pelo *File Transfer Protocol* (FTP). A pesquisa é realizada através de um motor de busca interno ou pela consulta directa das entradas presentes na base de dados. A submissão da informação para estas bases de dados é realizada pelos próprios investigadores ou por pessoal qualificado dentro da própria instituição. No primeiro caso, os dados podem ficar automaticamente disponíveis para consulta ou podem ter que passar, anteriormente, por um processo de validação (nas bases de dados curadas).

Múltiplas bases de dados tem sido desenvolvidas, ao longo do tempo, por várias instituições a nível global. As mais antigas foram iniciadas no princípio dos anos 60 do século passado e eram constituídas por sequências de proteínas. Mais tarde, nos anos 80, foram desenvolvidas as bases de dados dedicadas ao armazenamento de sequências de nucleótidos que estão actualmente instaladas nos servidores do NCBI, do EMBL/EBI e do DDBJ. Estas três instituições formam o *International Sequence Database Collaboration* e realizam entre si, diariamente, a transferência das novas sequências que lhes são submetidas. Desse modo a mesma informação é disponibilizada em triplicado, dividindo por cada instituição o peso do tráfego electrónico resultante dos milhões de pesquisas realizadas diariamente em todo o mundo [114, 115].

O NCBI é talvez o mais importante fornecedor de serviços de bioinformática à escala global. Criado em 1988 com o objectivo de desenvolver sistemas de informação para biologia molecular, foi alargando a sua gama de serviços, ao ponto de disponibilizar, na actualidade, dezenas de bases de dados e ainda ferramentas de análise úteis a toda a comunidade científica. A "face" mais conhecida é o "portal" *Entrez* (Figura 12) que permite a pesquisa integrada de informação sobre um conjunto de mais de 20 bases de dados. Estas incluem, entre outros, sequências de DNA e proteínas de várias fontes, taxonomia (*Taxonomy Browser*), genes (*Entrez genes*), genomas (*Entrez Genomes*), dados de expressão de genes (GEO), SNP's (dbSNP), estruturas de proteínas (*Molecular Modeling Database*, MMDB), informação sobre doenças (*Online Mendelian Inheritance in Man*, OMIM) e artigos de revistas científicas (PubMed). Um sistema de hiperligações permite navegar entre entradas de diferentes bases de dados, promovendo um estudo multidisciplinar do objecto pesquisado [116].

Figura 12 - Aspecto geral da página de entrada do Entrez do NCBI. Retirado de http://www.ncbi.nlm.nih.gov/gquery.

Para além de bases de dados, o site do NCBI disponibiliza igualmente várias ferramentas bioinformáticas úteis. A análise por alinhamentos de sequências de nucleótidos ou resíduos de aminoácidos pode ser realizada com o *Basic Local Alignment Search Tool*, BLAST (explicado mais à frente). O ORF *Finder* pesquisa uma sequência de DNA e determina a localização de cada *Open Reading Frame* (ORF), ou seja a série de codões na mesma grelha que se inicia com um codão de iniciação e termina num codão de terminação e que poderá corresponder a um gene. De entre as várias ferramentas relacionadas com genomas destaca-se o *Map Viewer*, que expõe visualmente mapas genómicos completos de vários organismos (actualmente, estão disponíveis 29 genomas) e incorpora hiperligações para entradas em várias bases de dados relacionadas com a secção do genoma que está exposta [116].

A base de dados GenBank (e a do EMBL e DDBJ) inclui todas as sequências de DNA e proteína disponíveis publicamente, com anotações descrevendo a informação biológica que cada entrada contém. Estas sequências são, na maioria dos casos, fragmentos simples e contíguos de DNA ou RNA. Os ficheiros presentes no GenBank estão agrupados em divisões; algumas destas divisões são baseadas na filogenia enquanto outras baseiam-se na tecnologia utilizada para gerar a informação sobre a sequência. Actualmente, todas as entradas do GenBank são geradas a partir de submissões directas de sequências pelos seus autores, que as enviam voluntariamente para ficarem publicamente disponíveis ou como parte do processo de publicação de artigos em revistas científicas. No entanto, as entradas submetidas não são revistas, excepto nos casos em que tal é solicitado, o que significa que existem erros e redundâncias várias em determinadas partes da base de dados.

Estas três bases de dados não curadas (GenBank, EMBL e DDBJ) são designadas de primárias e funcionam essencialmente como arquivo. Outras bases de dados são designadas de secundárias, pois obtêm a maioria das suas sequências a partir das anteriores e são curadas, ou seja, as suas entradas sofrem um processo de validação antes de serem disponibilizadas à comunidade. Um exemplo é o *Entrez Gene* (sucessor do LocusLink) do próprio NCBI. Cada entrada desta base de dados é curada e perfeitamente anotada e inclui a sequência de um gene conhecido de uma determinada espécie. Outros exemplos de bases de dados secundárias são a SWISS-PROT e a *Protein Information Resource* (PIR), dedicadas a sequências proteicas. Embora as bases de dados curadas possuam uma superior garantia de credibilidade em relação às primárias, o número de entradas que as constituem é normalmente várias ordens de grandeza inferior, dado que o processo de validação e anotação completa de uma sequência é um processo moroso. Por esta razão, uma vasta gama de sequências de grande utilidade para estudos bioinformáticos apenas se encontra nas bases de dados do GenBank, EMBL ou DDBJ. A sua utilização requer, em quaisquer condições, um acréscimo de precaução e a recolha de sequências das várias entradas semelhantes, no sentido de minimizar possíveis erros.

Embora estes repositórios globais possuam a quase totalidade dos fragmentos de DNA sequenciados, existem outras bases de dados dedicadas a organismos particulares. Cada projecto não comercial de sequenciação de um genoma possui geralmente uma página *Web* onde são disponibilizadas as sequências desse genoma. Um exemplo de base de dados especializada é a *Saccharomyces Genome Database* (SGD) pertencente ao

Stanford Human Genome Center. Esta base de dados possui uma interface muito simples que permite a pesquisa por nome do gene, informação do gene ou da proteína, clone, nome da sequência, nome do autor ou todo o texto.

Na Tabela 1 são apresentadas as instituições responsáveis por alguns dos projectos de sequenciação de genomas de fungos e o estado de desenvolvimento em que se encontram os trabalhos. Cada uma das suas páginas *Web* possui um esquema próprio de organização e pesquisa das entradas, o que dificulta, por vezes, a recolha de informação para várias espécies em simultâneo e induz o utilizador a tentar obter as sequências de que necessita através das bases de dados universais.

Tabela 1 – Instituições responsáveis por alguns projectos de sequenciação de genomas de fungos.

Espécie	Instituição responsável	Estado
A. fumigatus	*The Institute for Genomic Research*, TIGR (EUA)	Montagem
C. albicans	*Stanford University* (EUA)	Montagem
C. glabrata	*Genolevures Consortium* (França)	Completo
C. tropicalis	*Genolevures Consortium* (França)	Em progresso
C. neoformans	*The Institute for Genomic Research*, TIGR (EUA)	Completo
S bayanus	*Broad Institute* (EUA)	Montagem
S. cerevisiae	Colaboração de várias instituições mundiais	Completo
S. mikatae	*Broad Institute* (EUA)	Montagem
S. paradoxus	*Broad Institute* (EUA)	Montagem
S. pombe	*S. pombe European Sequencing Consortium*	Completo

Um caso particular de interface para bases de dados é o *Sequence Retrieval System* ligado ao EBI. Na página deste serviço, podem ser seleccionadas previamente as bases de dados (bibliotecas) de interesse. As bases de dados disponíveis incluem a de sequências de nucleótidos do EMBL ou suas subdivisões, Ensembl (genomas eucariótas anotados), SWISS-PROT, de estruturas de proteínas, de SNP's, de mutações ou mesmo de literatura, entre outras (Figura 13). De seguida o utilizador pode, através de uma única interface, formular pedidos de pesquisa complexos a todas as bases de dados em simultâneo.

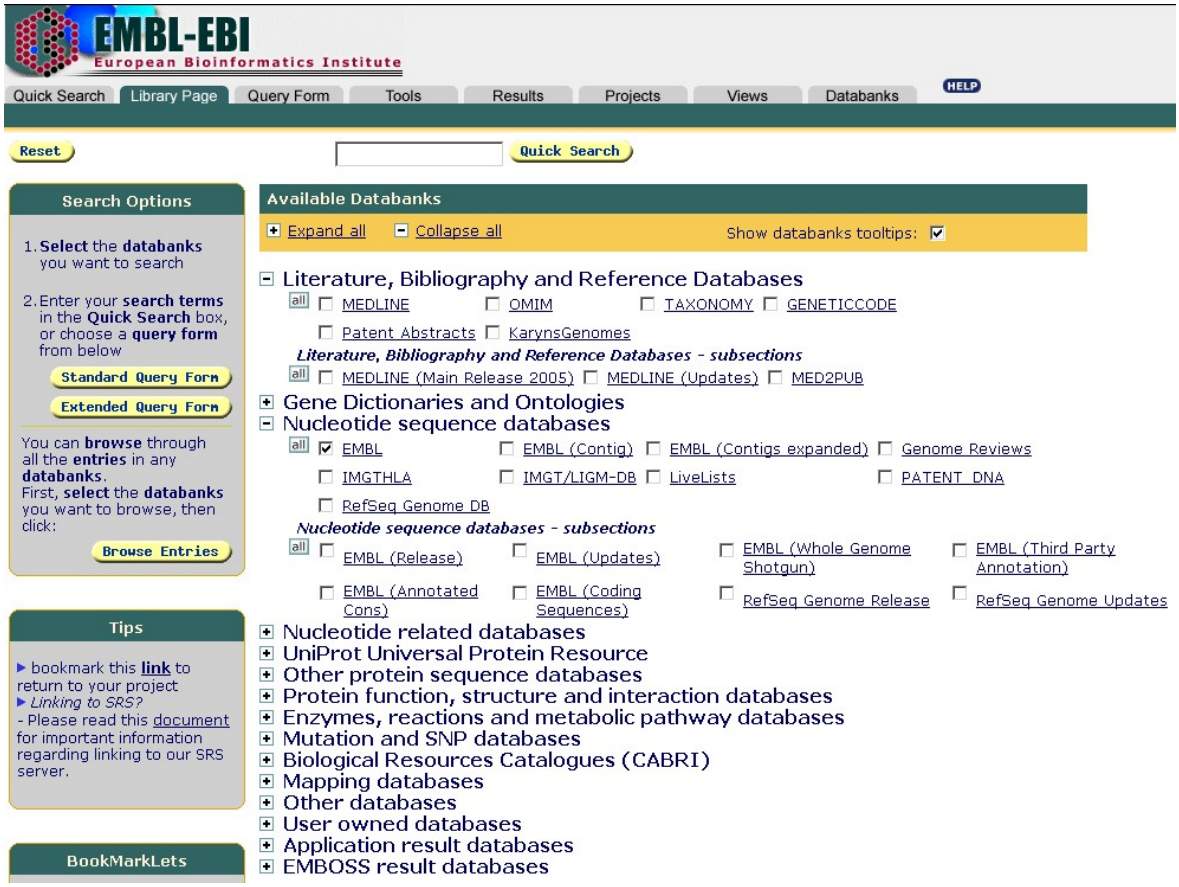

Figura 13 – Aspecto geral da página do SRS dedicada à selecção das bases de dados para pesquisa. São visíveis os vários grupos em que estas estão integradas. Dentro do grupo *Sequence* está seleccionada a base de dados EMBL. Retirado de http://srs.ebi.ac.uk.

4.1.1. Formatos de ficheiros de sequências

Um resultado de uma pesquisa numa base de dados de sequências biológicas em qualquer página *Web* pode ser extraído num ficheiro ASCII padrão. Este possui a própria sequência e uma série de anotações auxiliares. No entanto, conforme as suas origens, existem diferenças na presença ou ausência de certos caracteres ou palavras que indicam onde diferentes tipos de informação, incluindo a sequência, podem ser encontrados.

O formato mais simples de todos é o FASTA, que possui uma única linha de informações adicionais, o cabeçalho, iniciado pelo caracter "maior que" (>) seguido da sequência de nucleótidos ou da proteína nas linhas posteriores (Figura 14). Este formato permite um fácil tratamento da sequência primária de uma forma compreensível ao homem e aos computadores. No entanto é parco em anotações e a informação que possui no cabeçalho não está estruturada, o que dificulta, em certa medida, o seu aproveitamento em processos automáticos realizados por programas informáticos [114].

```
>gi|1502354
GAATTCGCAGAAATTGTTTTCTGTGCCATCAGTTCGACGGAACAGGTACGCGGTTATGGTGCGCATCTAA
TGAATCACTTAAAAGACTATGTTAGAAATACCTCGAACATAAAATATTTTTTGACATATGCAGATAATTA
CGCTATTGGATACTTTAAAAAGCAAGGCTTCACTAAAGAAATCACGTTGGATAAAAGTATATGGATGGGA
TATATTAAAGATTATGAAGGTGGTACGCTGATGCAATGTTCTATGTTACCAAGAATACGATATTTGGACG
CAGGTAAGATTCTATTATTACAAGAAGCGGCCCTGCGAAGAAAAATAAGAACGATTTCGAAATCGCATAT
TGTAAGGCCTGGTTTAGAGCAATTCAAAGACTTAAACAATATCAAACCGATTGATCCAATGACTATTCCT
GGCTTGAAAGAAGCCGGCTGGACTCCCGAGATGGATGCGTTGGCACAACGTCCCAAGCGTGGTCCACACG
ATGCAGCAATACAGAATATACTCACAGAGCTACAAAATCATGCAGCAGCTTGGCCCTTCTTACAACCCGT
TAATAAAGAGGAGGTCCCCGACTATTATGATTTTATCAAAGAGCCAATGGACTTGAGCACCATGGAAATA
AAATTAGAGAGCAACAAATATCAGAAGATGGAAGACTTCATATATGATGCCAGATTGGTGTTTAACAATT
GCCGAATGTACAATGGCGAGAATACGTCGTATTACAAGTATGCTAATAGGCTAGAGAAATTCTTCAATAA
TAAAGTAAAAGAAATACCTGAATATTCTCACCTTATTGATTAATGCGTAGAAGAAGCTTTTCCGCTACTA
TTCCTTTCGAAGAAGAAATAAATGTTTAGTACGGCGAGACGATGTGATCAATTGAGGTTATTTTACTACT
TTTCCTTTCATTTTTGTAAGGTTTTCTTTCTTTGTTAGTGTGACGTTGGTATTTACCTTTATGTAACTAT
```

Figura 14 – Pormenor de um registro em formato FASTA para uma sequência primária de nucleótidos. No cabeçalho, constituído por uma só linha, está indicado o código de acesso de uma sequência genómica do cromossoma VII de *S. cerevisiae*.

Outros formatos importantes estão associados às bases de dados que os utilizam. O Formato GenBank contém informação que descreve a sequência, incluindo referências bibliográficas, informação acerca da sua função, localização de mRNA's e regiões codificantes e posições de mutações. Esta informação está organizada em campos, assinalados por um identificador apresentado como a primeira palavra de cada linha. No campo "FEATURES" é incluído o subcampo "CDS" (*Coding Sequence*) que exibe a sequência de aminoácidos obtida pela tradução de ORF's potenciais. A sequência primária encontra-se entre os identificadores "ORIGIN" e "//" e possui um número em cada linha para permitir a localização visual de nucleótidos em posições específicas [115] (Figura 15).

```
LOCUS       SCCHRVII               10534 bp    DNA     linear   PLN 18-APR-2005
DEFINITION  S.cerevisiae genomic sequence from chromosome VII.
ACCESSION   X99228
VERSION     X99228.1  GI:1502354
KEYWORDS    6-phosphogluconate dehydrogenase; 6PGD2 gene; ENO1 gene; GCN5 gene;
            Mitochondrial carrier protein; PUP2 gene; transfer RNA.
SOURCE      Saccharomyces cerevisiae (baker's yeast)
  ORGANISM  Saccharomyces cerevisiae
            Eukaryota; Fungi; Ascomycota; Saccharomycotina; Saccharomycetes;
            Saccharomycetales; Saccharomycetaceae; Saccharomyces.
REFERENCE   1
  AUTHORS   Mazzoni,C., Ruzzi,M., Rinaldi,T., Solinas,F., Montebove,F. and
            Frontali,L.
  JOURNAL   Unpublished
REFERENCE   2  (bases 1 to 10534)
  AUTHORS   Mazzoni,C.
  TITLE     Direct Submission
  JOURNAL   Submitted (09-JUL-1996) C. Mazzoni, University of Rome 'La
            Sapienza', Cellular and Development Biology, Piazzale Aldo Moro 5,
            Rome, 00185, Italy
FEATURES             Location/Qualifiers
     source          1..10534
                     /organism="Saccharomyces cerevisiae"
                     /mol_type="genomic DNA"
                     /db_xref="taxon:4932"
                     /chromosome="VII"
     gene            <1..813
                     /gene="GCN5"
     CDS             <1..813
                     /gene="GCN5"
                     /codon_start=1
                     /protein_id="CAA67614.1"
                     /db_xref="GI:1502355"
                     /db_xref="GOA:Q03330"
                     /db_xref="UniProt/Swiss-Prot:Q03330"
                     /translation="EFAEIVFCAISSTEQVRGYGAHLMNHLKDYVRNTSNIKYFLTYA
                     DNYAIGYFKKQGFTKEITLDKSIWMGYIKDYEGGTLMQCSMLPRIRYLDAGKILLLQE
                     AALRRKIRTISKSHIVRPGLEQFKDLNNIKPIDPMTIPGLKEAGWTPEMDALAQRPKR
                     GPHDAAIQNILTELQNHAAAWPFLQPVNKEEVPDYYDFIKEPMDLSTMEIKLESNKYQ
                     KMEDFIYDARLVFNNCRMYNGENTSYYKYANRLEKFFNNKVKEIPEYSHLID"
```

Figura 15 – Pormenor de um registro em formato GenBank para uma sequência primária de nucleótidos. A sequência de nucleótidos primária não é visível.

O formato adoptado pelo NCBI para armazenar as entradas das suas várias bases de dados é o *Abstract Syntax Notation* (ASN.1). O ASN é uma linguagem descritiva de dados formais desenvolvido pela indústria informática. Por esse motivo, os dados podem ser facilmente acedidos por computadores. Este formato é muito estruturado e contém toda a informação presente noutros formatos, incluindo o GenBank. No entanto, a informação nele contida é mais difícil de ler, dado que está desenhada para o acesso através de programas informáticos [115].

O formato EMBL é similar ao formato GenBank, com pequenas modificações. Os identificadores são abreviados para duas letras, a sequência primária encontra-se entre os identificadores “SQ” e “//” e não é incluída a sequência de quaisquer produtos de tradução, os quais são por sua vez apresentados numa entrada separada da base de dados [115].

Existem vários outros formatos que são actualmente utilizados, incluindo os formatos SWISS-PROT, PIR e *Genetic Data Environment*. Uma ferramenta muito útil relacionada com esta multiplicação de formatos é o READSEQ, desenvolvido na Universidade de Indiana (EUA). Este programa consegue reconhecer um ficheiro com sequências de proteína ou DNA num de vários formatos e produz um novo ficheiro num formato alternativo.

4.2. Alinhamentos de pares de sequências – BLAST.

O método comparativo mais comum em bioinformática é o alinhamento de sequências. A sua construção pode ser baseada unicamente em duas, nos alinhamentos de pares de sequências (*pairwise*) ou em várias simultaneamente, nos alinhamentos múltiplos (ver secção seguinte). Nos últimos 30 anos, ao mesmo tempo que o número de sequências disponíveis nas bases de dados aumentou exponencialmente, foram igualmente desenvolvidos algoritmos mais rápidos e sofisticados que permitem a sua comparação por alinhamentos [114].

Os alinhamentos de pares de sequências tem um âmbito muito mais vasto do que a comparação entre duas sequências conhecidas e de tamanhos semelhantes. A utilização deste tipo de algoritmos permite a comparação de uma sequência contra todas as de uma base de dados inteira. Esta aplicação tem sido extensivamente utilizada em vários estudos de biologia molecular.

Ao comparar duas sequências de genes (ou proteínas) diferentes tenta-se determinar o grau de semelhança que existe entre elas. Se esse grau for suficientemente elevado é possível concluir que os genes são homólogos, ou seja, partilham pelo menos parte das suas histórias evolutivas, com a existência de um ancestral comum. As alterações que ocorreram durante o período de divergência evolutiva são traduzidas nas substituições, inserções e eliminações nas sequências dos seus nucleótidos. Se o método de alinhamento for ideal, os nucleótidos que estejam alinhados mas não sejam idênticos representam substituições. Regiões em que nucleótidos de uma sequência não correspondam a nada na outra podem ser interpretados como inserções numa sequência ou eliminações na outra. Nestes casos, são adicionados espaçamentos (*gaps*) representados por "–", na sequência sem nucleótidos [114].

Um elevado grau de semelhança entre as sequências de dois genes ou de duas proteínas indica que eles são homólogos(as) e possuem funções semelhantes. No entanto, esta última dedução tem que ser acompanhada de estudos bioquímicos que a confirmem, pois um dos genes pode ter-se adaptado a condições diferentes e alterado a sua função.

O número de algoritmos de alinhamentos que é possível aplicar para qualquer problema é extraordinariamente grande. No entanto, a grande maioria baseia-se em contribuições incrementais para cada par de nucleótidos, ou *gap*-nucleótido, do alinhamento. São adicionados valores positivos quando os nucleótidos alinhados são iguais ou idênticos e valores negativos nos casos em que se observam substituições ou *gaps*.

Os algoritmos mais simples que são utilizados no alinhamento de sequências são as matrizes do tipo *Dot Plot*. Neste método, duas sequências são dispostas numa matriz bidimensional. Os nucleótidos de uma são dispostos no eixo dos *xx* e os da outra no eixo do *yy*. Em cada elemento da matriz é colocado um ponto se os nucleótidos coincidirem ou é deixado em branco se não coincidirem (Figura 16).

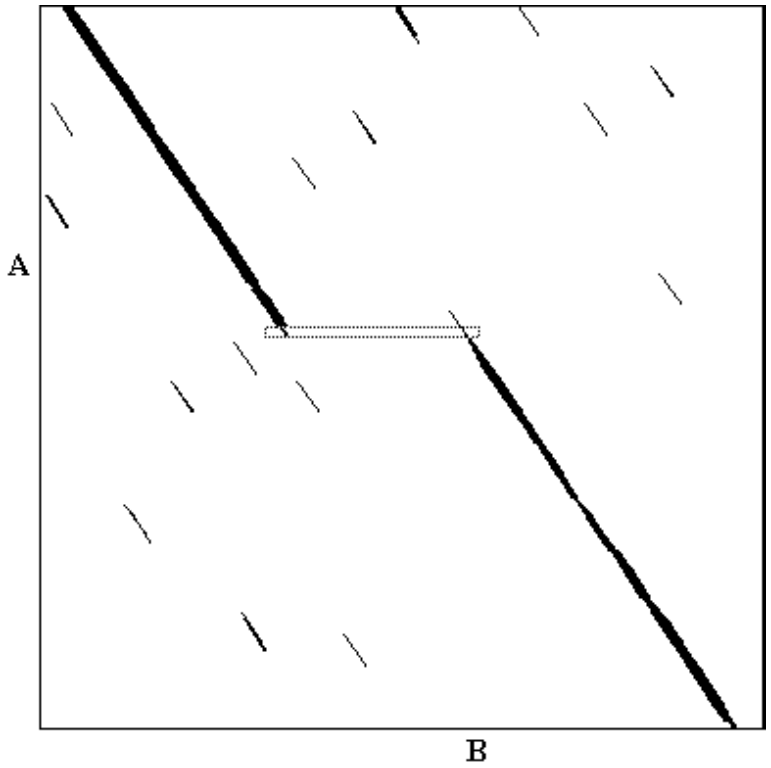

Figura 16 – Exemplo de uma matriz *Dot Plot*. A descontinuidade na diagonal, visível ao centro da matriz, indica que nas posições centrais da sequência B existe uma secção (caixa a tracejado) que não encontra correspondência na sequência A. Durante o processo evolutivo, poderá ter ocorrido inserção de nucleótidos na sequência B ou a sua eliminação na sequência A. Adaptado de http://bioinformatics.weizmann.ac.il.

O objectivo deste método é uma visualização gráfica simples das regiões de identidade entre as duas sequências. As regiões de semelhança aparecem como séries de pontos na diagonal. Algumas diferenças entre as duas sequências aparecem com assinaturas características na matriz. Um exemplo é a presença de diagonais invertidas que correspondem a inversões no sentido de uma das sequências. Estas séries diagonais podem posteriormente ser unidas numa alinhamento com *gaps* introduzidos sempre que há uma descontinuidade.

Outros métodos de alinhamentos, tais como *Dynamic programming*, desenvolvido por Needleman e Wunsch em 1970 e refinados em 1981 por T. Smith e M. Waterman, são tidos como matematicamente óptimos porque determinam sempre o melhor alinhamento possível. No entanto, são algoritmos lentos, pois requerem um grande número de passos computacionais, e podem criar alinhamentos com pouco significado a nível biológico.

Para se conseguir alinhar, de forma rápida, uma sequência com todas as de uma base de dados foram criados novos algoritmos de pesquisa. Destes métodos de alinhamentos locais rápidos destacam-se o FASTA, desenvolvido por W. Pearson (1988) e o BLAST desenvolvido por Altshul e colegas (1990) [117] e refinado posteriormente (1997) [118]. Os seus modos de funcionamento são semelhantes, com a utilização de métodos estatísticos e heurísticos simplificados e a segmentação das sequências em pequenas "palavras".

O BLAST é a ferramenta de alinhamentos mais rápida e mais utilizada em estudos com sequências de nucleótidos ou aminoácidos. Entre a comunidade científica, o seu algoritmo é aceite como padrão. A sua utilização é predominantemente realizada na pesquisa de bases de dados que contêm milhões de sequências, como é caso da GenBank. Na pesquisa de bases de dados, a operação básica é alinhar sistematicamente a sequência em estudo (*query*) com cada sequência (*subject*) da base de dados.

Quando uma pesquisa é realizada, o BLAST cria uma tabela com todas as palavras, isto é, secções da sequência com um determinado tamanho (indicado pelo utilizador). De seguida varre a base de dados na procura dessas palavras. Quando encontra a palavra exacta, executa rapidamente uma matriz para calcular a qualidade desse alinhamento parcial. Se o valor encontrado for acima de um determinado limiar *T* (especificado no

algoritmo), o algoritmo inicia uma tentativa de encontrar um alinhamento local máximo com um valor (*score*) igual ou superior ao valor de um segundo limiar, *S*. Para isso, inicia um processo repetitivo de extensão do alinhamento para a direita e para a esquerda com acumulação dos valores incrementais correspondentes aos *matches* (nucleótidos iguais), *mismatches* (nucleótidos diferentes) e introdução de *gaps*, que encontre. Em regiões em que os *matches* sejam escassos, o valor acumulado começa a diminuir. À medida que o valor decresce torna-se menos provável que atinja o limiar *S*. Métodos heurísticos são então utilizados para determinar se o algoritmo deve continuar a estender o alinhamento ou se deve passar ao próximo [114, 119].

O limiar *S* anteriormente referido é inversamente proporcional ao limiar *E* definido pelo utilizador antes da pesquisa. Este parâmetro afecta os resultados obtidos no final da pesquisa. Se *S* for muito elevado, a pesquisa é mais restringente e apenas os melhores alinhamentos serão apresentados. Cada alinhamento apresentado nos resultados possui um valor designado de *bit score*, que não é mais que o valor acumulado, anteriormente referido, e que é, obrigatoriamente, igual ou superior a *S*. O *bit score* é um indicador da qualidade do alinhamento. Quanto maior for, melhor é o alinhamento. A cada alinhamento é ainda anexado outro valor, o *Expect value* (*E-value*) que fornece uma indicação da importância do alinhamento e repercute ainda o tamanho da base de dados e o sistema de valorização utilizado. O *E-value* descreve o número de *hits* que se espera obter, apenas ao acaso, quando se pesquisa uma base de dados de um determinado tamanho. Quanto mais pequeno for (mais se aproximar de zero), mais significativo é o alinhamento. Um *E-value* de 0.05 significa que esta semelhança tem 5 probabilidades em 100 de ocorrer por acaso. Embora, à primeira vista, possa parecer significativo, pode não representar um resultado importante a nível biológico. Será necessário analisar o próprio alinhamento para inferir o seu real valor [119].

Embora este seja o método geral de funcionamento do algoritmo BLAST, existem múltiplas combinações de parâmetros que podem ser utilizadas para diferentes finalidades. A página do BLAST nos servidores de NCBI (http://ncbi.nlm.nih.gov/BLAST), agrupa as variedades de BLAST pelo tipo de pesquisa: nucleótidos, proteína, nucleótidos traduzidos e genomas. Para cada combinação de interesse, existe uma página, com um conjunto de parâmetros predefinidos, onde se pode introduzir a sequência *query* a pesquisar [120]. Na

Figura 17 é apresentada a página dedicada à pesquisa de sequências de nucleótidos em bases de dados de nucleótidos, com os parâmetros predefinidos. Na caixa denominada "*Search*" é introduzida a sequência *query*, em formato FASTA ou sem qualquer informação adicional.

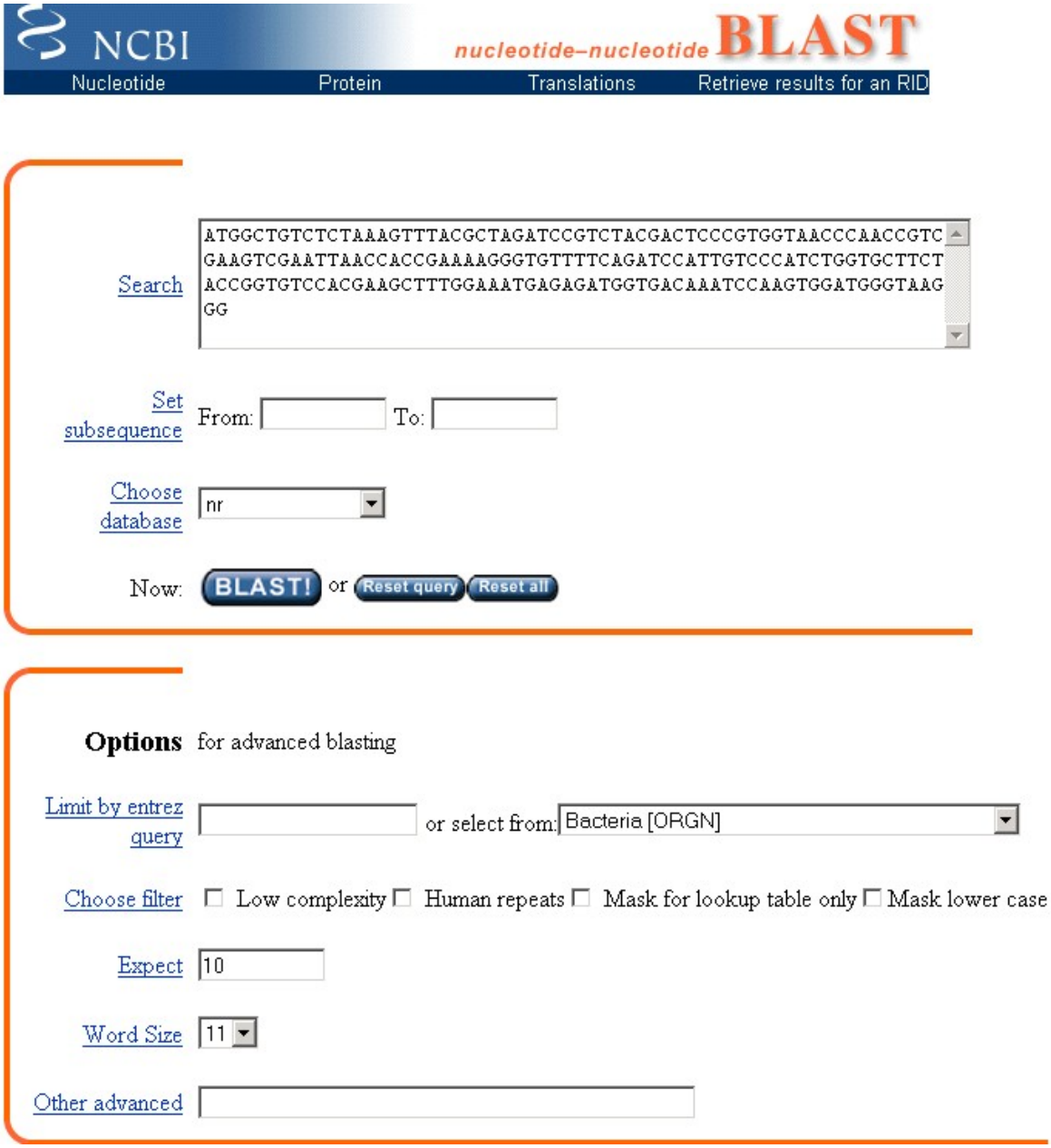

Figura 17 - Página do BLAST para pesquisa de sequências de nucleótidos em bases de dados de nucleótidos. Entre os parâmetros predefinidos estão o *E value* (10) e o *Word size* (11). A base de dados seleccionada é de nucleótidos "não-redundante" e está limitada a sequências pertencentes a espécies de bactérias.

Um dos parâmetros mais importantes nestes processos é a escolha da base de dados. A universal ("não-redundante", "nr") contém biliões de bases sequenciadas até ao

momento, incluindo todo o GenBank, o EMBL e o DDBJ. No entanto, é possível pesquisar unicamente determinadas secções desta base de dados, como o genoma humano, sequências de bactérias, sequências de fungos, ou muitas outras classificações taxonómicas.

O BLAST não pesquisa directamente os ficheiros de texto em formato GenBank. Em vez disso, as sequências necessitam ser formatadas antes de entrarem para as bases de dados do BLAST. Neste processo, cada entrada é dividida e dois ficheiros são criados; um contendo apenas a sequência e o outro a descrição dessa sequência. São estes ficheiros que o algoritmo pesquisa. Esta formatação da base de dados é obrigatória em qualquer pesquisa que o BLAST realize, quer seja online ou offline [119].

Para escolher qual a melhor opção para uma determinada pesquisa específica é aconselhado seguir as indicações do *Program Selection Guide* nas páginas *Web* de ajuda do BLAST. Aí estão indicadas quais as melhores opções, baseadas na natureza da sequência *query*, na finalidade da pesquisa e no tipo de base de dados pretendida.

Os resultados de uma pesquisa com o BLAST são apresentados no formato tradicional de um relatório. Este consiste em três partes distintas: 1- o cabeçalho, que contém informação acerca da sequência *query* e da base de dados pesquisada (Figura 18) e pode ainda exibir um gráfico sumário (Figura 19); 2- descrições de cada sequência da base de dados para a qual foi encontrada semelhança com a sequência *query* (Figura 20); 3- os próprios alinhamentos encontrados entre a sequência *query* e as sequências da base de dados (Figura 21). Cada sequência encontrada na base de dados é denominada de *Hit*, enquanto que cada alinhamento representa um *High-scoring Segment Pair* (HSP). É possível existir mais do que um HSP por cada *Hit*, desde que sejam referentes a diferentes secções.

BLASTN 2.2.10 [Oct-19-2004]

Reference:
Altschul, Stephen F., Thomas L. Madden, Alejandro A. Schäffer, Jinghui Zhang, Zheng Zhang, Webb Miller, and David J. Lipman (1997), "Gapped BLAST and PSI-BLAST: a new generation of protein database search programs", Nucleic Acids Res. 25:3389-3402.

RID: 1118947561-15157-108296152791.BLASTQ1

Query=
(182 letters)

Database: All GenBank+EMBL+DDBJ+PDB sequences (but no EST, STS, GSS,environmental samples or phase 0, 1 or 2 HTGS sequences)
3,206,819 sequences; 14,237,488,286 total letters

If you have any problems or questions with the results of this search please refer to the **BLAST FAQs**

Taxonomy reports

Figura 18 – Cabeçalho de um relatório de BLAST. A primeira linha fornece informação acerca do tipo de programa (neste caso BLASTN), a versão (2.2.10) e a data de distribuição (19 de Outubro de 2004). De seguida são apresentados a referência bibliográfica para o artigo que descreve o BLAST, o RID, informação sobre a sequência *query* e sobre a base de dados pesquisada. Finalmente aparece uma hiperligação para outro relatório com um formato alternativo que apresenta os resultados obtidos com base na classificação taxonómica das espécies às quais as sequências pertencem.

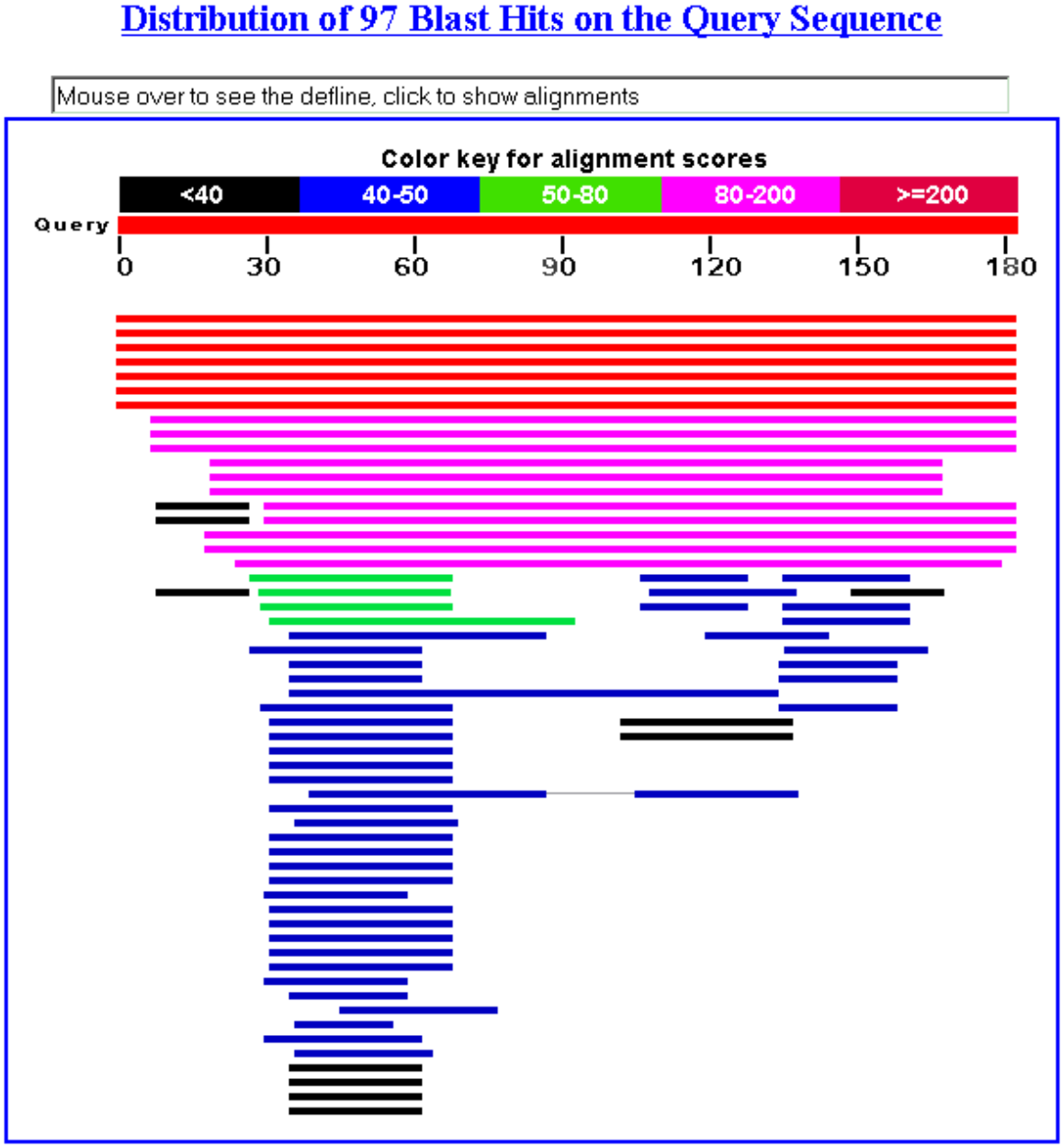

Figura 19 – Gráfico sumário dos resultados do BLAST. A sequência *query* é representada por uma barra vermelha numerada no topo da figura. Os *hits* são representados pelas barras inferiores que se encontram alinhadas. As barras que se encontram nas posições superiores representam os melhores alinhamentos. O seu tamanho e posição no gráfico são consequências directas do alinhamento. Ao sobrepor o rato sobre uma das barras é mostrada a descrição de cada *hit*.

Sequences producing significant alignments:	Score (Bits)	E Value	
gi|1502354|emb|X99228.1|SCCHRVII S.cerevisiae genomic sequence f	361	4e-97	G
gi|1323461|emb|Z73039.1|SCYGR254W S.cerevisiae chromosome VII re	361	4e-97	G
gi|171454|qb|J01322.1|YSCENOA S.cerevisiae enolase gene (clone p	361	4e-97	
gi|41614696|qb|U00027.2|YSCH9986 Saccharomyces cerevisiae chromo	321	3e-85	
gi|171456|qb|J01323.1|YSCENOB Yeast (S. cerevisiae) enolase gene	321	3e-85	G
gi|49526570|emb|CR380955.1| Candida glabrata strain CBS138 chrom	289	1e-75	
gi|50289856|ref|XM_447360.1| Candida glabrata CBS138, CAGL0I0248	289	1e-75	G
gi|49640134|emb|CR382121.1| Kluyveromyces lactis strain NRRL ...	180	8e-43	
gi|37693128|emb|AJ586240.1| Kluyveromyces lactis eno gene for en	180	8e-43	
gi|50302928|ref|XM_451402.1| Kluyveromyces lactis NRRL Y-1140, K	180	8e-43	G
gi|170860|qb|L04943.1|YSAENO1A Candida albicans enolase (ENO1) g	167	1e-38	
gi|170862|qb|L10290.1|YSAENOLAS Candida albicans enolase (ENO1)	167	1e-38	
gi|170864|qb|M93712.1|YSAENOLASE Candida albicans (clone lambda-	167	1e-38	
gi|49657202|emb|CR382139.1| Debaryomyces hansenii chromosome ...	143	2e-31	
gi|50427088|ref|XM_462151.1| Debaryomyces hansenii CBS767, DEHA0	143	2e-31	G
gi|45016181|qb|AE016818.1| Ashbya gossypii (= Eremothecium go...	127	1e-26	
gi|47074123|ref|NM_210505.1| Eremothecium gossypii AER294Cp (AER	127	1e-26	G
gi|30314939|qb|AF382946.1| Rhodotorula mucilaginosa enolase mRNA	117	1e-23	
gi|46108927|ref|XM_381522.1| Gibberella zeae PH-1 strain PH-1; N	58.0	8e-06	G
gi|37702654|qb|U82438.1|CHU82438 Cladosporium herbarum enolase g	54.0	1e-04	
gi|467659|emb|X78226.1|CHCLAH2 C.herbarum ClaH2 mRNA for enolase	54.0	1e-04	
gi|22035896|emb|AJ496792.1|ASI496792 Anisakis simplex mRNA for e	52.0	5e-04	
gi|32992620|dbj|AK107411.1| Oryza sativa (japonica cultivar-g...	48.1	0.008	
gi|5832087|emb|AL116871.1|CNS01DIN Botrytis cinerea strain T4 cD	46.1	0.031	
gi|41324904|emb|BX927150.1| Corynebacterium glutamicum ATCC 1...	46.1	0.031	
gi|42602314|dbj|BA000036.3| Corynebacterium glutamicum ATCC 1303	46.1	0.031	
gi|49073515|ref|XM_400971.1| Ustilago maydis 521, UM03356.1 pred	46.1	0.031	
gi|40806811|qb|AY499570.1| Cryphonectria parasitica enolase (Eno	46.1	0.031	
gi|33285257|qb|AC145782.1| Pan troglodytes BAC clone CH251-48...	44.1	0.12	
gi|3152298|emb|Y17298.1|CEY17298 Cunninghamella elegans mRNA for	44.1	0.12	
gi|762851|qb|L36343.1|ZMOHISHA Zymomonas mobilis imidazole ac...	44.1	0.12	
gi|56542470|qb|AE008692.1| Zymomonas mobilis subsp. mobilis ZM4,	44.1	0.12	
gi|49532994|dbj|BS000608.1| Pan troglodytes chromosome Y clon...	44.1	0.12	

Figura 20 – Pormenor da parte superior de um resumo descritivo de cada *hit* numa única linha. Cada linha é constituída por 5 campos: (1) o identificador da entrada da sequência *hit* na base de dados sob a forma de uma hiperligação; (2) uma breve descrição da sequência *hit*, a sua definição. Esta linha está truncada para manter a linha compacta; (3) o valor do *bit score*. Os valores mais altos encontram-se no topo da lista; (4) O valor do *E-value*. Os valores mais pequenos encontram-se no topo da lista; (5) hiperligação de algumas sequências *hit* para a entrada correspondente no *Entrez Gene* (G). O número de *hits* aqui apresentados pode ser definido na altura da pesquisa ou da formatação dos resultados.

```
>gi|37693128|emb|AJ586240.1|  Kluyveromyces lactis eno gene for enolase,
           sub-strain 2359/152
          Length=2856

 Score =  180 bits (91),  Expect = 8e-43
 Identities = 154/175 (88%), Gaps = 0/175 (0%)
 Strand=Plus/Plus

Query  8     TCTCTAAAGTTTACGCTAGATCCGTCTACGACTCCCGTGGTAACCCAACCGTCGAAGTCG  67
             ||||||||||||||| |||||||||||||| || |||||||||||||| || ||||| |
Sbjct  1140  TCTCTAAAGTTTACGCCAGATCCGTCTACGATTCTCGTGGTAACCCAACTGTTGAAGTTG  1199

Query  68    AATTAACCACCGAAAAGGGTGTTTTCAGATCCATTGTCCCATCTGGTGCTTCTACCGGTG  127
             |||| ||||||||||||||  |||||||||||||| |||||||||| || || ||||
Sbjct  1200  AATTGACCACCGAAAAGGGTACTTTCAGATCCATTGTTCCATCTGGTGCCTCCACTGGTG  1259

Query  128   TCCACGAAGCTTTGGAAATGAGAGATGGTGACAAATCCAAGTGGATGGGTAAGGG  182
             | ||||||||| | ||| |||||||||||||||||| || ||| | ||||||||
Sbjct  1260  TTCACGAAGCTCTAGAATTGAGAGATGGTGACAAATCTAAATGGTTAGGTAAGGG  1314
```

Figura 21 – Pormenor da secção de alinhamentos de um relatório do BLAST. O alinhamento é precedido pelo identificador da sequência *hit* na base de dados (hiperligação), pela descrição completa e pela sua extensão, em pares de bases (2856). A linha seguinte contém o *bit score* (180) e o *E-value* (8e-43). Posteriormente aparece o número e percentagem de nucleótidos iguais entre as sequências do alinhamento (154 em 175, 88%), o número e percentagem de gaps (0) e em qual das cadeias complementares são encontradas as sequências (ambas cadeia *plus*). Finalmente é apresentado o alinhamento com a sequência *query* por cima da sequência encontrada ("*Sbjct*"). Os números à esquerda e à direita referem-se às posições dos nucleótidos. O número de alinhamentos apresentados nesta secção do relatório pode ser definido na altura da pesquisa ou da formatação dos resultados.

Este formato do relatório está desenhado para apresentar os resultados de forma a serem facilmente visualizados pelo utilizador. Exemplos disso são o gráfico sumário que oferece uma perspectiva visual e muito global dos resultados ou a descrição de cada *hit* numa só linha. Ainda para conveniência de utilização, é repetida a apresentação de alguma informação (por exemplo, *E-value* e *bit score*) e são integradas várias hiperligações para outros recursos online. No entanto, é possível realizar o seu parsing de modo automático (ver secção 4.4).

O aspecto global do relatório de uma pesquisa com o BLAST pode ser alterado através da sua formatação. Mesmo após uma primeira visualização, o relatório pode ser de novo pedido aos servidores do BLAST através do identificador único atribuído a cada pesquisa, o *Request Identifier* (RID).

Para além de o aceder através das páginas Web do NCBI, existem outras formas de utilizar o BLAST. Uma delas é instalando um pacote informático, designado de Standalone BLAST, num computador local. Os executáveis para várias plataformas informáticas, incluindo o MS-Windows, podem ser extraídos das páginas FTP do NCBI (ftp://ftp.ncbi.nih.gov/blast/executables/). É também possível extrair algumas bases de

dados em formato *flatfile* (por exemplo, GenBank ou FASTA) ou já formatadas (ftp://ftp.ncbi.nih.gov/blast/db/). Com a utilização desta ferramenta, a pesquisa deixa de estar dependente das flutuações nos servidores de Internet, pode ser realizada em bases de dados personalizadas e de uma forma muito mais rápida. No entanto, as bases de dados poderão necessitar de ser regularmente actualizadas e os relatórios de BLAST não integram o gráfico sumário (Figura 19) nem as hiperligações presentes no seu análogo online.

4.3. Alinhamentos múltiplos de sequências – Clustal.

Para comparar mais do que duas sequências simultaneamente é necessário recorrer a alinhamentos múltiplos. O objectivo deste processo é conseguir expor o máximo de semelhança entre todas as sequências em análise ou entre grupos de sequências. Ao proceder a um alinhamento múltiplo está-se implicitamente a considerar que algumas ou todas as sequências são homólogas, ou seja, possuam um ancestral comum mesmo que muito distante. De outro modo, qualquer semelhança que se detecte entre as sequências, não terá qualquer significado biológico.

Os alinhamentos múltiplos são particularmente úteis na análise de proteínas. Quando uma sequência proteica acaba de ser sequenciada, um dos objectivos imediatamente considerados é a sua caracterização. A sequência é então pesquisada contra as sequências depositadas nas bases de dados e sobre as quais já se possua algum grau de informação adicional. Se for encontrada mais do que uma sequência semelhante, procede-se a um alinhamento múltiplo entre todas. A utilização destes algoritmos permite encontrar padrões idênticos entre as várias sequências, os quais poderão fornecer pistas para futuros estudos. Este é normalmente o primeiro passo na determinação da estrutura tridimensional e função de proteínas [114].

A utilização de alinhamentos múltiplos também se pode revelar importante na comparação de sequências de DNA. Por exemplo, a determinação do grau de semelhança entre genes permite retirar conclusões acerca da filogenia molecular das espécies. A um nível mais prático, um alinhamento múltiplo pode facilitar a escolha de *primers* para amplificar por PCR-*Multiplex* uma determinada região em várias sequências/espécies.

Um alinhamento múltiplo de sequências de DNA ou proteína é criado quando os elementos constituintes (nucleótidos ou resíduos) de uma sequência são alinhados com os

de pelo menos outra sequência. Um alinhamento requer normalmente a inserção de *gaps* nas sequências para se puderem dispor os elementos semelhantes nas mesmas colunas (Figura 22).

```
gi|9581744|emb|CAC00532.1|      MAITIVSVRARQIFDSRGNPTVEADVKLSDGYLARAAVPSGASTGIYEAL
gi|602253|gb|AAD04187.1|        MAATIQSVKARQIFDSRGNPTVEVDVFCSDGTFARAAVPSGASTGVYEAL
gi|16271|emb|CAA41114.1|        -MATITVVKARQIFDSRGNPTVEVDIHTSNGIKVTAAVPSGASTGIYEAL
gi|45477377|gb|AAS66001.1|      -MATITVVKARQIFDSRGNPTVEVDIHTSNGVKVTAAVPSGASTGIYEAL
gi|6321693|ref|NP_011770.1|     --MAVSKVYARSVYDSRGNPTVEVELTTEKGVFR-SIVPSGASTGVHEAL
                                   ::  * **.::*********.::  ..*    : ********::***

gi|9581744|emb|CAC00532.1|      ELRDGG-SDYLGKGVSKAVENVNIIIGPALVGK--DPTDQVGIDNFMVQQ
gi|602253|gb|AAD04187.1|        ELRDGG-SYYLGKGVSKAVNNVNSVIGPALIGK--DPTAQTEIDNFMVQQ
gi|16271|emb|CAA41114.1|        ELRDGG-SDYLGKGVSKAVGNVNNIIGPALIGK--DPTQQTAIDNFMVHE
gi|45477377|gb|AAS66001.1|      ELRDGG-SDYLGKGVSKAVGNVNNIIGPALIGK--DPTQQTAIDNFMVHE
gi|6321693|ref|NP_011770.1|     EMRDGDKSKWMGKGVLHAVKNVNDVIAPAFVKANIDVKDQKAVDDFLIS-
                                *:***. * ::**** :** *** :*.**::    * . *  :*:*::

gi|9581744|emb|CAC00532.1|      LDGTVNEWGWCKQKLGANAILAVSLAVCKAGAHVKGIPLYEHIANLAGNK
gi|602253|gb|AAD04187.1|        LDGTKNEWGWCKQKLGANAILAVSLAVCKAGASIKRIPLYQHIANLAGNK
gi|16271|emb|CAA41114.1|        LDGTQNEWGWCKQKLGANAILAVSLAVCKAGAVVSGIPLYKHIANLAGNP
gi|45477377|gb|AAS66001.1|      LDGTQNEWGWCKQKLGANAILAVSLAVCKAGAVVSGIPLYKHIANLAGNP
gi|6321693|ref|NP_011770.1|     LDGTAN-----KSKLGANAILGVSLAASRAAAAEKNVPLYKHLADLSKSK
                                **** *     *.********.****..:*.*  . :***:*:*:*: .

gi|9581744|emb|CAC00532.1|      N--LVLPVPAFNVINGGSHAGNKLAMQEFMILPVGASSFKEAMKMGAEVY
gi|602253|gb|AAD04187.1|        Q--LVLPVPAFNVINGGSHAGNKLAMQEFMILPTGAASFKEAMKMGVEVY
gi|16271|emb|CAA41114.1|        K--IVLPVPAFNVINGGSHAGNKLAMQEFMILPVGAASFKEAMKMGVEVY
gi|45477377|gb|AAS66001.1|      K--IVLPVPAFNVINGGSHAGNKLAMQEFMILPVGASSFKEAMKMGVEVY
gi|6321693|ref|NP_011770.1|     TSPYVLPVPFLNVLNGGSHAGGALALQEFMIAPTGAKTFAEALRIGSEVY
                                    ***** :**:*******. **:***** *.** :* **:::* ***
```

Figura 22 – Alinhamento entre cinco sequências proteicas.

Existem vários métodos e algoritmos para produzir alinhamentos múltiplos de sequências. Os métodos progressivos criam um alinhamento múltiplo começando pelas sequências com mais semelhanças, e de seguida vão adicionando progressivamente as sequências ou grupos menos semelhantes. As relações entre as sequências são moduladas numa árvore filogenética criada em simultâneo. Este tipo de construção de alinhamentos emprega métodos heurísticos, não é exaustivo e pode não encontrar o alinhamento óptimo [115]. A série Clustal é um exemplo de algoritmos baseados neste método.

O Clustal inicia o processo de criação de alinhamentos múltiplos pelo alinhamento de todos os pares de sequências possíveis. Para cada um desses alinhamentos atribui um *score*, através de uma matriz [n x n]. De seguida, ordena as sequências de acordo com a capacidade demonstrada de se alinharem par a par e, simultaneamente, utiliza os *scores* calculados desses alinhamentos para construir uma árvore filogenética. Posteriormente, inicia a construção do alinhamento múltiplo pelo par com mais semelhanças entre si e vai adicionando progressivamente as outras sequências, guiado pela árvore filogenética [121].

Esta árvore é disponibilizada após este processo ter terminado como complemento ao alinhamento criado.

Estes programas bioinformáticos têm vindo a ser constantemente melhorados e actualizados com o desenvolvimento de novos algoritmos e alterações na interface de utilização. O princípio orientador responsável por esta evolução tem sido a criação de programas robustos, simples e capazes de realizar alinhamentos precisos num curto espaço de tempo [122]. O ClustalW inclui unicamente o programa, executável numa linha de comandos, com algoritmo necessário à realização dos alinhamentos. Este programa pode ser instalado localmente pelos utilizadores nas suas máquinas ou ser acedido através de certas páginas *Web*, como por exemplo no servidor ClustalWWW do EBI.

Mais recentemente foi desenvolvido o ClustalX, com uma interface gráfica [123]. Embora os resultados produzidos com as versões mais recentes de ambos os programas sejam os mesmos, o ClustalX permite ao utilizador uma melhor apreciação visual dos alinhamentos.

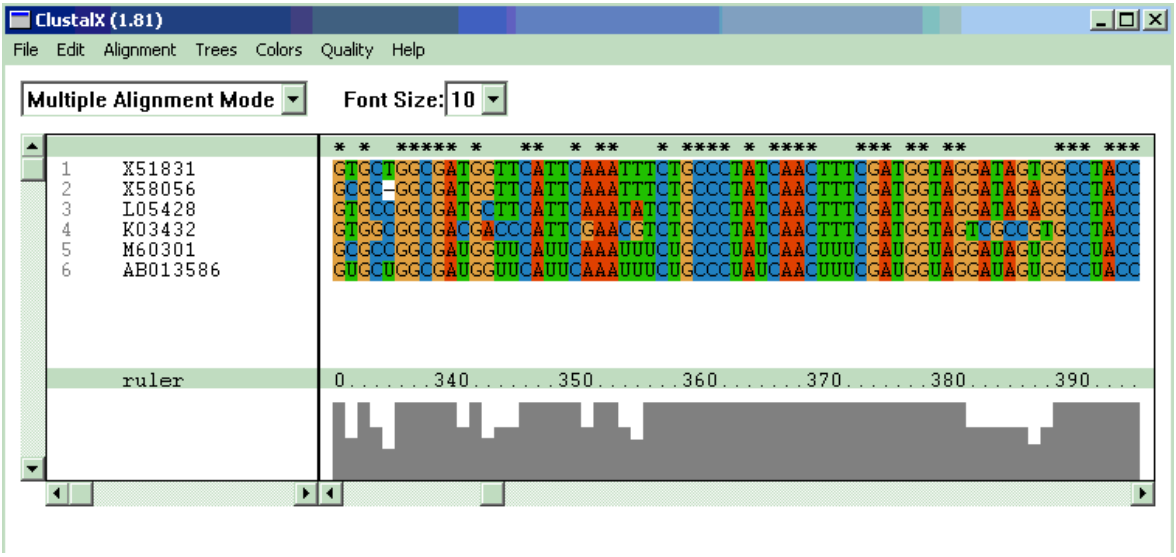

Figura 23 – Pormenor de um alinhamento de 6 sequências de nucleótidos visualizado no ClustalX (v.1.81). Cada nucleótido tem uma cor: A – vermelho; C – azul; G – amarelo; T – verde. O gráfico de barras por baixo da régua de numeração indica o grau de conservação de cada posição entre as 6 sequências.

O ClustalX exibe o alinhamento múltiplo numa janela com barra de deslizamento e todos os parâmetros são disponibilizados em menus. Nos próprios alinhamentos cada nucleótido ou resíduo possui uma cor específica e as colunas conservadas são evidenciadas

por um gráfico de barras (Figura 23). Este programa é fácil de instalar, é *user-friendly* e funciona numa vasta gama de plataformas informáticas, MS-Windows incluído.

4.4. Programação informática – linguagem PERL.

A análise de um problema biológico com recurso a ferramentas bioinformáticas requer, na maioria dos casos, a repetição sistemática de certos procedimentos. O cálculo da temperatura de fusão de uma sonda de DNA através de uma fórmula termodinâmica é facilmente executável manualmente com recurso a uma calculadora ou folha de cálculo. No entanto, se for necessário realizar o mesmo procedimento para milhares de sondas de tamanhos variáveis, o tempo despendido será incomensurável e provavelmente serão cometidos alguns erros de cálculo. Poderá, também, ser necessário submeter sequências ao servidor de BLAST do NCBI e extrair os números de acesso de cada *hit*. Se o número de sequências for elevado, chega-se facilmente à conclusão de que a sua realização manual é impraticável.

Para resolver estes problemas, semelhantes aos de outras actividades profissionais ou de investigação, é imprescindível recorrer a programação informática. Várias linguagens têm vindo a ser aplicadas nos estudos bioinformáticos. De entre todas, destaco a linguagem de programação PERL (*Practical Extraction and Report Language*).

O PERL é dotado de determinadas características técnicas que facultam ao programador uma capacidade excepcional de manipular e integrar ficheiros de dados. Estas qualidades são especialmente evidentes no tratamento de dados textuais, tais como as sequências de DNA ou proteína. A linguagem PERL é vastamente utilizada em aplicações Web seguindo as normas CGI (*Common Gateway Interface*). O PERL tem uma vasta comunidade de entusiastas que desenvolvem aplicações modulares e as disponibilizam livremente. Foram desenvolvidas adaptações que lhe permitem correr nos principais sistemas operativos. Para MS-Windows foi desenvolvido o pacote informático *Open-Source* ActivePerl pela ActiveState Corp [124].

O PERL tem como desvantagens a morosidade quando comparado com C ou C++; a relativa falta de aproveitamento de memória RAM, que influencia o tamanho dos conjuntos de dados que podem ser analisados simultaneamente; e algumas soluções pouco intuitivas no desenho da linguagem, que podem dificultar a estratégia de resolução de certos problemas [124].

O PERL é uma linguagem de *scripting*, ou seja, o código é escrito num texto com a descrição, em linguagem formal, dos passos distintos a serem executados e o ficheiro resultante não é compilado. Quem executa estes passos é o interpretador que está acessível em qualquer directório do computador. Na plataforma MS-Windows o interpretador é o programa executável perl.exe que vem incluído no pacote ActivePerl. A execução dos *scripts*, bem como a introdução de eventuais parâmetros adicionais, pode ser realizado directamente na linha de comandos. No caso de um *script* de PERL que obedeça às normas CGI, pode ser executado sobre um servidor *Web*.

Uma das vantagens das distribuições de PERL é o facto de serem *Open-Source*. Deste modo, qualquer elemento da comunidade de programadores pode depositar o código que desenvolveu, e que considere útil, em repositórios públicos. Neste domínio, o repositório mais abrangente é o *Comprehensive Perl Archive Network*, CPAN (http://www.cpan.org). Determinados projectos, desenvolveram-se ao ponto de formarem organizações estruturadas de programadores e verificadores. Na área da bioinformática, o projecto mais significativo é o Bioperl (http://www.bioperl.org), uma colaboração entre biólogos, bioinformáticos e informáticos que se expandiu ao longo dos últimos 10 anos. O resultado deste projecto é uma biblioteca de módulos disponíveis para tratar e manipular informação biológica.

Uma das principais motivações do projecto Bioperl é focar-se numa solução em que os seus componentes sejam reutilizados e partilhados de modo a evitar a duplicação de esforços. Para isso, os módulos de Bioperl facilitam determinadas tarefas recorrentes nos estudos bioinformáticos. Com a sua utilização, poucas linhas de código são suficientes para realizar tarefas que, de outra forma, envolveriam dias de trabalho [125].

O Bioperl é capaz de executar análises e processar resultados de programas como o BLAST e o ClustalW. A manipulação de ficheiros em vários formatos e de origens distintas é facilitada pelo reconhecimento automático das suas características. Alguns dos seus módulos permitem mesmo a conversão entre formatos de ficheiros de sequência. Outras aplicações, para além da análise e anotação de sequências, tratam de filogenia, estrutura de proteínas e referências bibliográficas. Dado que algumas soluções já se encontram desenvolvidas noutras linguagens informáticas, como C, Java ou Python, o

Bioperl permite a interoperabilidade, ou seja, a invocação dessas ferramentas dentro dos próprios scripts de PERL [125].

O Bioperl é desenhado através de uma metodologia orientada a objectos. Este facto permite criar módulos genéricos, claros e reutilizáveis para representar estruturas de dados e operações típicas das ciências biológicas. Ao separar os componentes em grupos lógicos, tais como sequências, alinhamentos e bases de dados, é possível adicionar atributos a um módulo específico sem alterar necessariamente o resto da biblioteca. Esta separação é um dos aspectos essenciais na programação orientada a objectos e permite produzir componentes genéricos com uma interface estável para o programador.

E 5. Objectivos deste projecto.

Este trabalho tem como objectivos principais o desenvolvimento de metodologias de desenho de *chips* de DNA para diagnosticar infecções originadas por fungos patogénicos e ao mesmo tempo, a sua aplicação e validação no desenho de alguns desses *chips*. Para esta finalidade, é necessária a realização sequencial de determinados procedimentos parciais.

Serão desenvolvidos sistemas locais bioinformáticos de selecção de sondas, tendo como base a linguagem de programação PERL e algoritmos de alinhamentos de sequências, como o BLAST e o Clustal. O universo de análise destes sistemas será constituído, no mínimo, por sequências homólogas representativas do conjunto de espécies que se pretendem diagnosticar em cada *chip* de DNA. O resultado final será apresentado sob a forma de sequências de DNA específicas para cada uma das espécies. No processamento serão tidos em conta parâmetros termodinâmicos, como a temperatura de fusão e a formação de estrutura secundária.

Para a validação dos resultados produzidos nos sistemas locais, será desenvolvido um método de pesquisa das referidas sondas em bases de dados online universais, nomeadamente a instalada nos servidores do NCBI.

A aplicação de algumas das metodologias desenvolvidas será utilizada no desenho de *chips* de DNA de diagnóstico que incidirão nas espécies *Aspergillus fumigatus*, *Candida albicans*, *Candida glabrata*, *Candida tropicalis*, *Cryptococcus neoformans*, *Saccharomyces bayanus*, *Saccharomyces cerevisiae*, *Saccharomyces mikatae*, *Saccharomyces paradoxus* e *Schizosaccharomyces pombe*. As primeiras cinco espécies serão estudadas devido à sua importância infecciosa. A inclusão do conjunto de *Saccharomyces* é tido como um teste, de cariz filogenético, à capacidade discriminatória dos sistemas de selecção. A selecção de sondas específicas para estas espécies será realizada sobre as suas sequências de rDNA, que serão previamente extraídas de bases de dados online e submetidas a um tratamento.

Capítulo II: Metodologias Bioinformáticas utilizadas no *design* de um *chip* de diagnóstico molecular.

1. Escolha das espécies a diagnosticar.

O primeiro passo no *design* de um chip de diagnóstico é a escolha das espécies a ser identificadas. Neste estudo, foram incluídas espécies com base nos seus níveis de virulência, no quadro clínico, e na proximidade filogenética.

2. Download das sequências dos genes.

2.1. Pesquisa nas bases de dados online.

A pesquisa e download das sequências de genes de rRNA foi realizada nas páginas do *The European ribosomal RNA database*, *Sequence Retrieval System* (SRS) e *National Center for Biotechnology Information* (NCBI).

Da base de dados "*The European ribosomal RNA database*" foram importadas algumas das sequências referentes às espécies de fungos pretendidas (*Aspergillus fumigatus*, *Candida albicans*, *Candida glabrata*, *Candida tropicalis*, *Cryptococcus neoformans*, *Saccharomyces bayanus*, *Saccharomyces cerevisiae*, *Saccharomyces mikatae*, *Saccharomyces paradoxus* ou *Schizosaccharomyces pombe*). A base de dados encontra-se dividida em rRNA da Subunidade Grande do ribossoma (*Large Subunit*, LSU) e rRNA da Subunidade Pequena (*Small Subunit*, SSU). Dentro de cada divisão encontra-se uma lista de espécies ordenadas segundo a sua classificação taxonómica. Para cada espécie em estudo foram directamente extraídas todas as sequências disponíveis.

No SRS, foi seleccionada a base de dados de sequências de nucleótidos do *European Molecular Biology Laboratory – European Bioinformatics Institute* (EMBL-EBI), que é actualizada diariamente com os mais recentes resultados de sequenciações. Através da *Extended query form*, encontraram-se as sequências de interesse para cada espécie. Os campos Molecule; Description e Organism Name foram preenchidos com os dados necessários, respectivamente, genomic DNA; genes ou secção de genes de rRNA pretendidos (rRNA, ribossomal RNA, ITS1, ITS2, 18S, 16S, 26S ou 28S) e nome da espécie. Os termos introduzidos nos campos foram combinados com o operador booleano "AND". Após uma validação manual do conteúdo das entradas obtidas nos resultados, procedeu-se ao *download* das sequências como texto ASCII no formato FASTA.

Para as espécies das quais não se encontrou a sequência genómica completa dos genes de rRNA, recorreu-se à base de dados GenBank instalada nos servidores do NCBI, para as tentar completar. As sequências foram igualmente guardadas no formato FASTA.

Embora as bases de dados de sequências de nucleótidos do EMBL e do GenBank (bem como do *DNA Data Bank of Japan*) estejam permanentemente a partilhar informação, as diferenças nas interfaces e motores de pesquisa originam ligeiras discrepâncias no conjunto de entradas apresentadas como resultados.

2.2. Escolha de sequências representativas.

Com as sequências extraídas das bases de dados online foi realizada a montagem (*assembling*) da sequência codificante completa dos genes de rRNA de cada uma das 10 espécies. Este processo foi realizado recorrendo às zonas sobrepostas entre duas sequências adjacentes (Figura 24).

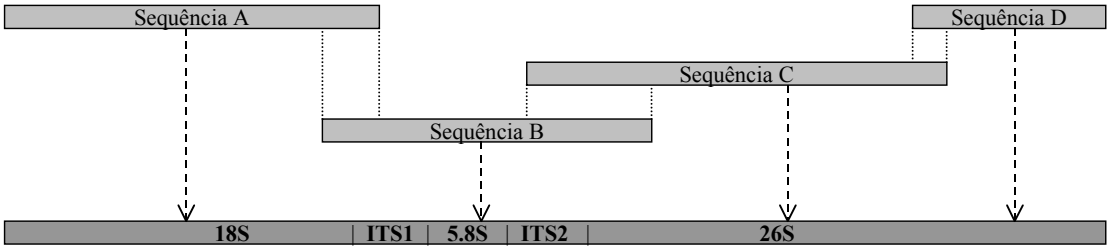

Figura 24 – Processo de montagem das sequências completas dos genes de rRNA a partir de sequências parciais extraídas das bases de dados online. As sequências A, B, C e D são parcialmente sobreponíveis nos seus extremos. A sequência final é o resultado da montagem dessas 4 sequências.

2.3. Anotação de *mismatches*.

Devido à redundância das bases de dados online, é comum encontrar mais do que uma entrada (sequência) para o mesmo gene. É igualmente usual que, entre algumas dessas sequências, existam diferenças na cadeia de nucleótidos.

Figura 25 – Pormenor de um alinhamento múltiplo de sequências equivalentes (do mesmo gene, SSU, da mesma espécie, *Cryptococcus neoformans*), mas que apresentam nucleótidos diferentes em duas posições – assinaladas com 0. Estas diferenças resultam de erros de sequenciação ou de diferenças intraespecíficas.

Durante o processo de montagem, sempre que duas (ou mais) sequências com diferentes nucleótidos para a mesma posição (ou a presença de mais nucleótidos numa delas) foram encontradas, a posição em causa foi identificada com a letra "N" do código IUPAC que representa qualquer base. No exemplo da Figura 25, estão alinhadas duas sequências de SSU de *Cryptococcus neoformans* com diferenças em duas posições. Numa delas está presente "A" (em neoformans_SSU_a) ou "G" (neoformans_SSU_b), enquanto que na outra existe "T" (neoformans_SSU_a) ou "C" (neoformans_SSU_b). Na sequência final foi colocado um "N" em cada uma das posições. Desta forma evitou-se que, posteriormente, se construissem sondas sobre zonas com diferenças intraespecíficas ou mal sequenciadas.

3. Construção da Base de Dados.

O conjunto das sequências de genes construídas na secção anterior foi usado como base de dados sobre a qual todas as pesquisas posteriores foram realizadas.

Quando a ferramenta de procura utilizada foi o BLAST, a base de dados foi constituída apenas por estas sequências (cadeia sense). Este algoritmo procurou os padrões de homologia na sequência da base de dados mas também na sequência da cadeia *antisense* (inversa e complementar). Antes de iniciar qualquer pesquisa com o BLAST, foi necessário formatar a base de dados, recorrendo ao comando formatdb (incluído no pacote

de instalação do Standalone BLAST) a partir da linha de comandos do MS-Windows (Figura 26). Nesta formatação, foi indicada a localização do ficheiro a ser formatado (-i C:\rRNA\database\database.txt) que é constituído por nucleótidos ("-p F") e. Os restantes argumentos não declarados foram assinalados com a opção predefinida.

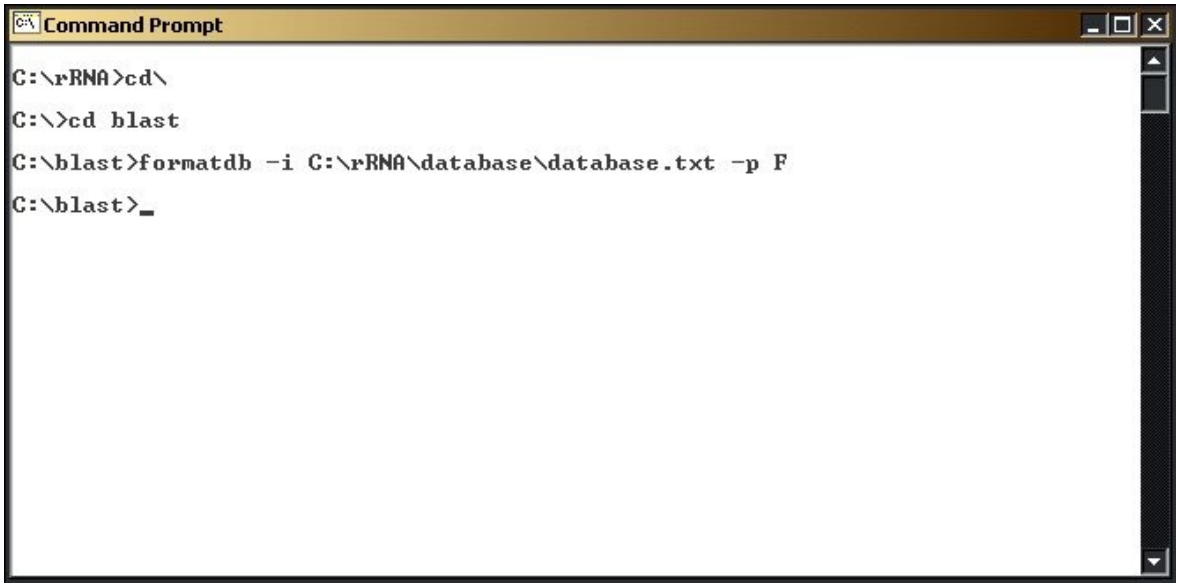

Figura 26 – Formatação da base de dados "database.txt" na linha de comandos do MS-Windows 2000.

Quando a base de dados foi utilizada numa pesquisa realizada por outro algoritmo de pesquisa, foi necessário adicionar a cadeia inversa e complementar de todas as suas sequências.

4. Sistemas locais de selecção de sondas.

Para determinar as sondas a colocar no chip, foram usados sistemas locais (em oposição a online), integrados em *scripts*, baseados em programação informática na linguagem PERL e em algoritmos matemáticos. A execução destes programas, bem como a introdução dos argumentos necessários foi igualmente realizada na linha de comandos do MS-Windows.

Estes sistemas têm, de um modo geral, um processo de funcionamento semelhante: recebem como *input* as sequências completas (ou zonas manualmente escolhidas) dos genes de rRNA, escolhem janelas sucessivas (Figura 27) de "n" nucleótidos e caracterizam-nas de acordo com parâmetros específicos.

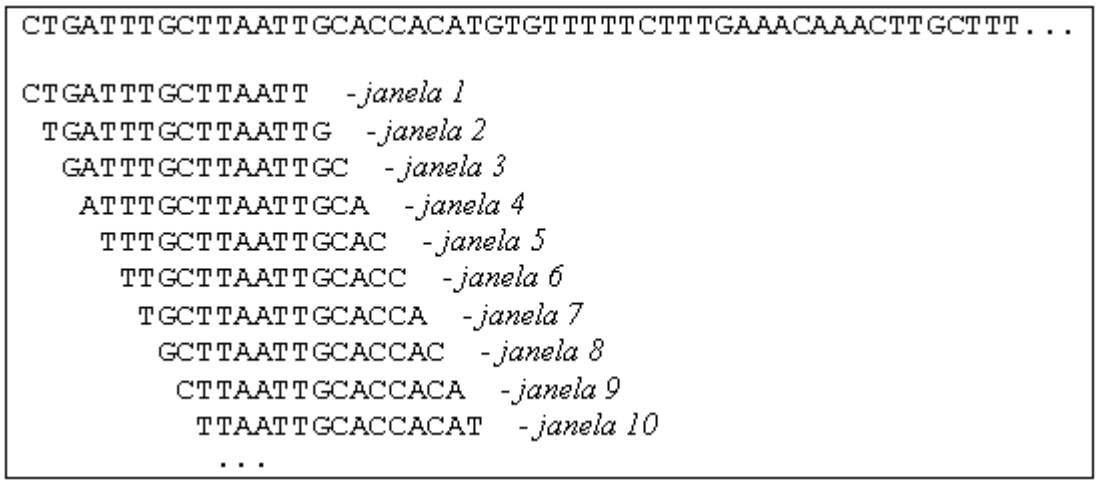

Figura 27 – Representação esquemática do processo base de selecção de janelas sucessivas (de 15 nucleótidos) de uma sequência. Estas janelas serão de seguida pesquisadas individualmente na base de dados utilizando diferentes algoritmos.

A caracterização de cada janela de nucleótidos permitiu determinar a possibilidade de cada uma funcionar como sonda específica para uma dada espécie num chip de diagnóstico. Este processo foi automatizado e baseou-se na indicação das variações entre sequências de espécies diferentes (recorrendo a diferentes parâmetros de acordo com o sistema adoptado), recorrendo a algoritmos matemáticos de alinhamentos, integrados em *scripts* de PERL.

Nestes sistemas recorreu-se a vários algoritmos de alinhamentos de sequências sendo que numa primeira aproximação foi utilizado o BLAST, fornecido pelo NCBI, dada a sua fiabilidade, facilidade de utilização e personalização.

Foram usados vários métodos de *design* do chip, que se diferenciam pelo tamanho das sondas seleccionadas e pela definição de "sequência específica". Em relação ao tamanho, foram desenhadas sondas grandes, de 50 nucleótidos, e sondas mais pequenas de apenas 15 nucleótidos específicos.

Para seleccionar as sondas maiores, fez-se a pesquisa com o algoritmo BLAST e analisou-se de seguida a informação produzida. Os resultados de cada pesquisa de uma sequência *query* numa base de dados com o *Standalone* BLAST foram apresentados sob a forma de um ficheiro de texto - um relatório. Os 3 métodos seguidos para seleccionar as sondas de 50 nucleótidos basearam-se no seu *parsing*, com níveis crescentes de aprofundamento da informação retirada (secções 4.1, 4.2, 4.3).

O *parsing* de um relatório de BLAST designa a extracção minuciosa de qualquer porção da informação detalhada presente neste ficheiro de texto. Os dados extraídos (texto descritivo, sequências ou valores numéricos) podem pertencer a qualquer uma das três partes de um relatório: cabeçalho, descrição de cada sequência *hit* em linhas únicas e alinhamentos.

Em relação às sondas pequenas, utilizaram-se outros algoritmos e tentaram encontrar-se sequências, com a obrigatoriedade de haver diferenças interespecíficas em determinadas posições da sonda (secção 4.4).

4.1. Sondas grandes baseadas no *Hit Score*.

Os relatórios do BLAST apresentam as classificações dos melhores alinhamentos entre a sequência *query* e as sequências da base de dados (*Hits*) sob a forma de valores numéricos – *Hit Score* e *E value*. O *Hit Score* (também designado de *bit score*) é tanto maior quanto mais significativo for o alinhamento, enquanto que o *E value* é um indicador inversamente proporcional.

As janelas com maior probabilidade de funcionar como sondas específicas são aquelas que para além de só se encontrarem numa única espécie são também significativamente diferentes das sequências das outras espécies. Deste modo, na primeira aproximação para seleccionar um conjunto de sondas para o chip, o objectivo foi encontrar janelas de sequência de nucleótidos cujos relatórios (resultantes da pesquisa com o BLAST na base de dados local) tivessem *Hit Scores* inespecíficos (encontrados noutras espécies) o mais pequenos possíveis.

Tomando cada janela como sequência *query* e fazendo pesquisas sistemáticas com o BLAST na base de dados local, obtiveram-se múltiplos relatórios (tantos quantas as janelas). Este processo foi realizado pelo *script* jblast2_4.pl (Anexo 5).

Os relatórios foram de seguida ordenados de acordo com o *Score* do seu maior *Hit* inespecífico. Quando existiram dois ou mais relatórios com o mesmo valor, recorreu-se aos *Scores* dos segundos maiores *Hits* (se existirem) e assim sucessivamente até se obter uma classificação de todos os relatórios. Este *parsing* de resultados foi realizado pelo *script* parsereport1j.pl (Anexo 6).

A Figura 28 é uma esquematização do modo de funcionamento destes dois programas.

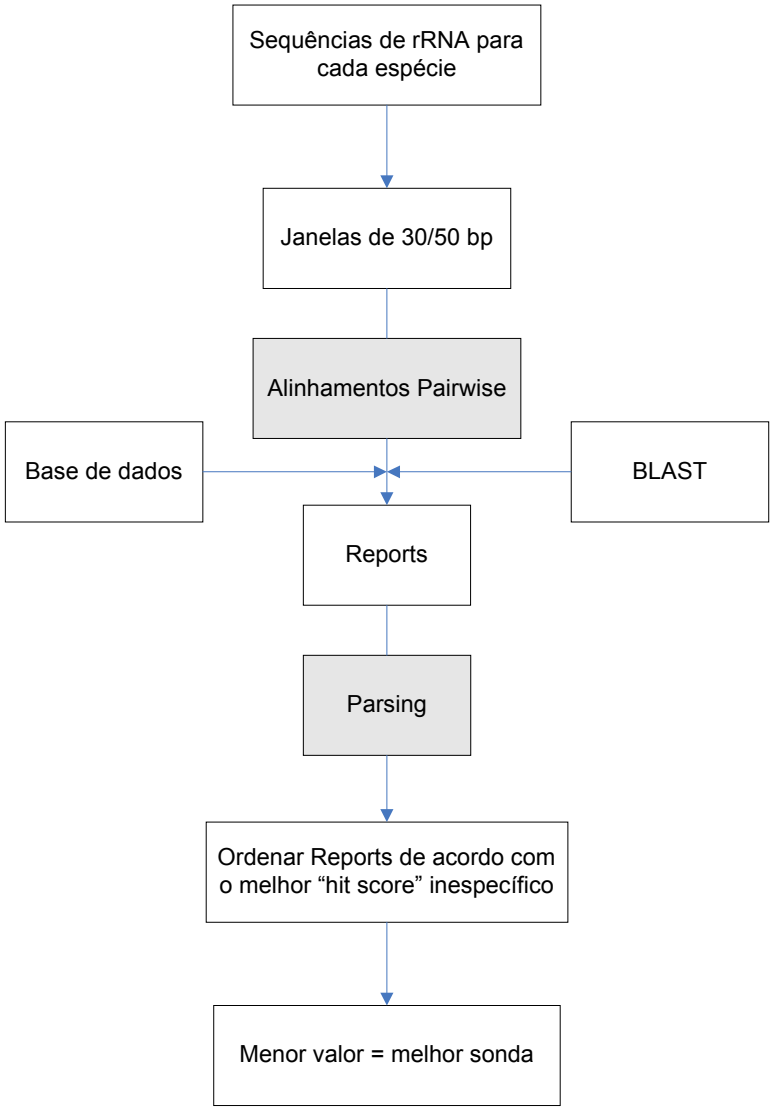

Figura 28 – Fluxograma do processo de escolha de sondas baseadas nos *Hit Score* inespecíficos dos relatórios do BLAST. O *script* jblast2_4.pl origina um relatório por cada janela por pesquisa na base de dados com o BLAST. O *script* parsereport1j.pl faz o *parsing* desses ficheiros e ordena as janelas correspondentes.

Os resultados finais do *parsing* foram exibidos num ficheiro de texto que apresenta todos os relatórios (cada um correspondendo a cada janela) ordenados (os primeiros são os mais indicados para gerar sondas). Para cada relatório foi também indicado os valores de todos os seus *Scores* inespecíficos.

4.2. Sondas grandes baseadas nos *matches* entre *hitstring* e *querystring*.

Tendo em conta que a classificação numérica nos relatórios do BLAST (*Scores*) não é uma representação inequívoca dos alinhamentos, a segunda aproximação ao *parsing*

focou-se na configuração do próprio *Hit* – a relação entre a sequência *Query* e a sequência *Hit*.

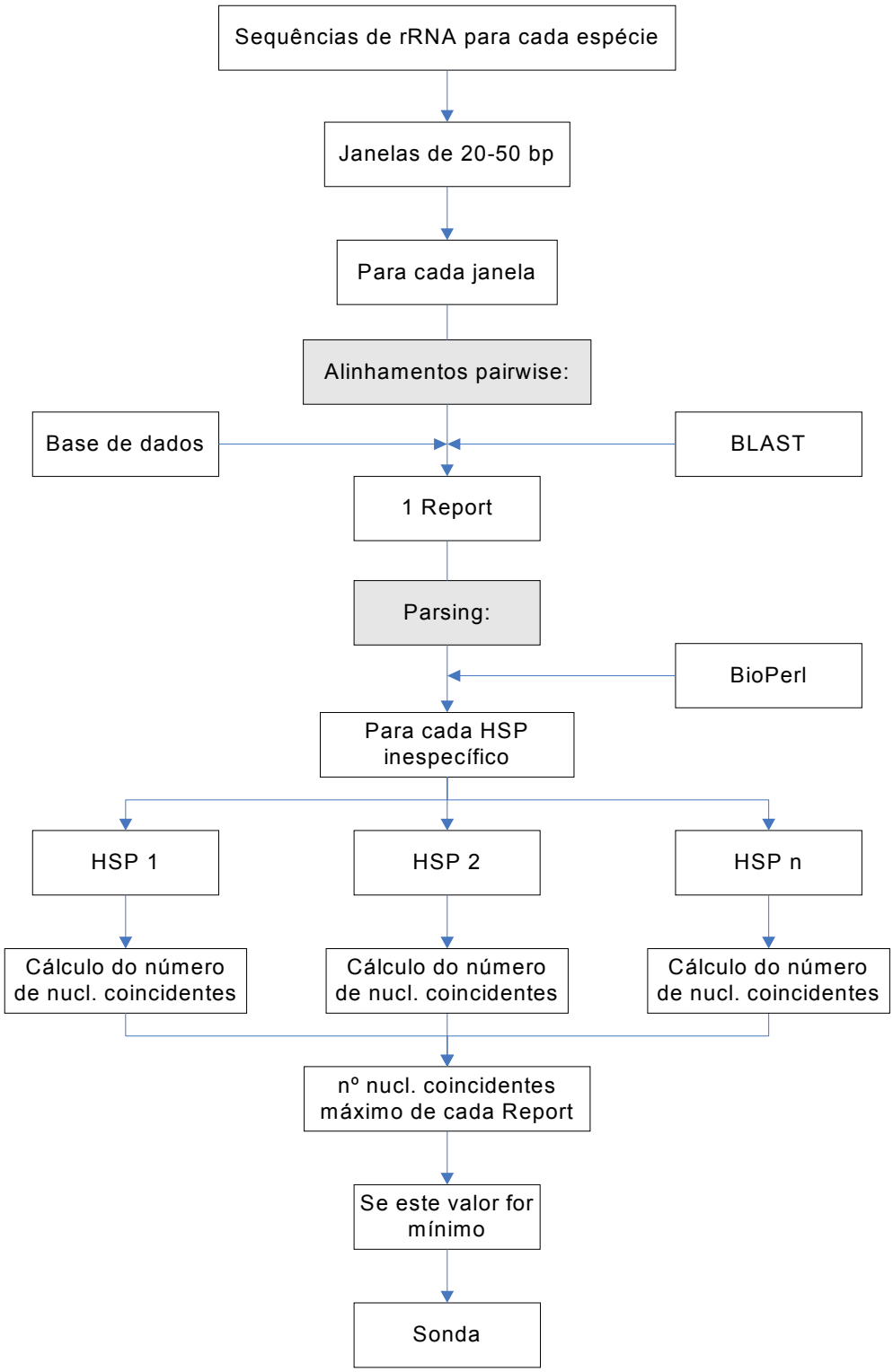

Figura 29 – Fluxograma do modo de funcionamento do *script* jblast4b.pl. Este programa realiza a pesquisa sistemática das janelas na base de dados (Alinhamentos *pairwise*) realizando de seguida o *parsing* e a ordenação dos relatórios com base na comparação das sequências *Query* com as sequências *Subject* dos HSP dos relatórios do BLAST.

O *script* jblast4b.pl (Anexo 7), representado na Figura 29, utilizou-se para fazer quer a pesquisa das janelas na base de dados (com o BLAST) quer o *parsing* dos relatórios resultantes, por uma questão de sistematização. A pesquisa com o BLAST foi realizada nos moldes anteriores, com a produção de um relatório típico por cada janela pesquisada. O *parsing* foi efectuado de um modo mais elaborado e recorrendo a mais dados fornecidos em cada relatório.

A todos os alinhamentos dos HSP de *Hits* inespecíficos apresentados foram contados o número de nucleótidos iguais entre a sequência *Query* e a sequência *Subject*. Nucleótidos diferentes foram automaticamente descontados (Figura 30). Este processamento foi realizado no *script* recorrendo a módulos do Bioperl.

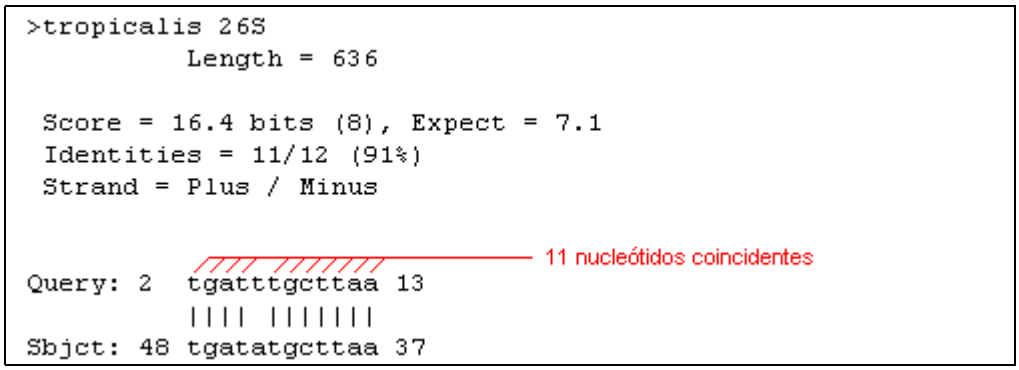

Figura 30 – Pormenor de um relatório de BLAST com contagem (integrada no *script* jblast4b.pl) dos nucleótidos iguais entre a sequência *query* e a sequência *subject* de um alinhamento de um HSP. A extensão de ambas as sequências é de 12 bases, mas existe um *mismatch* na quinta posição do alinhamento resultando em 11 nucleótidos coincidentes.

Após contar todos os alinhamentos de um relatório, todos os seus valores foram armazenados num ficheiro de texto a ele associado. De seguida todos os relatórios foram ordenados - de acordo com o *match* com o maior número de nucleótidos coincidentes de cada um. As janelas (sequências pesquisadas) a que correspondem os relatórios com valores mais baixos (*matches* inespecíficos mais pequenos) são aquelas com maiores possibilidades de funcionar como sondas.

Os resultados finais foram apresentados num ficheiro HTML com hiperligações para todos os ficheiros criados e também para as janelas que foram pesquisadas pelo BLAST (e também para a sequência completa que deu origem às janelas). Deste modo, foi

possível aceder facilmente aos resultados mais significativos e às sequências que os originaram, no momento de escolher as sondas para o chip.

4.3. Sondas grandes baseadas nas Tm dos *hits* inespecíficos.

O próximo passo após contar o número de nucleótidos coincidentes nas sequências do alinhamento (secção anterior), foi adicionar, no *parsing* dos relatórios, uma análise qualitativa a esses mesmos nucleótidos.

O *script* jblast3d.pl (Anexo 8), representado na Figura 31, é semelhante ao anterior jblast4b.pl, mas em vez de apenas contar os nucleótidos coincidentes entre sequência *Query* e sequência *Subject* (de todos os HSP de todos os relatórios), calculou, também, a temperatura de hibridação de cada alinhamento.

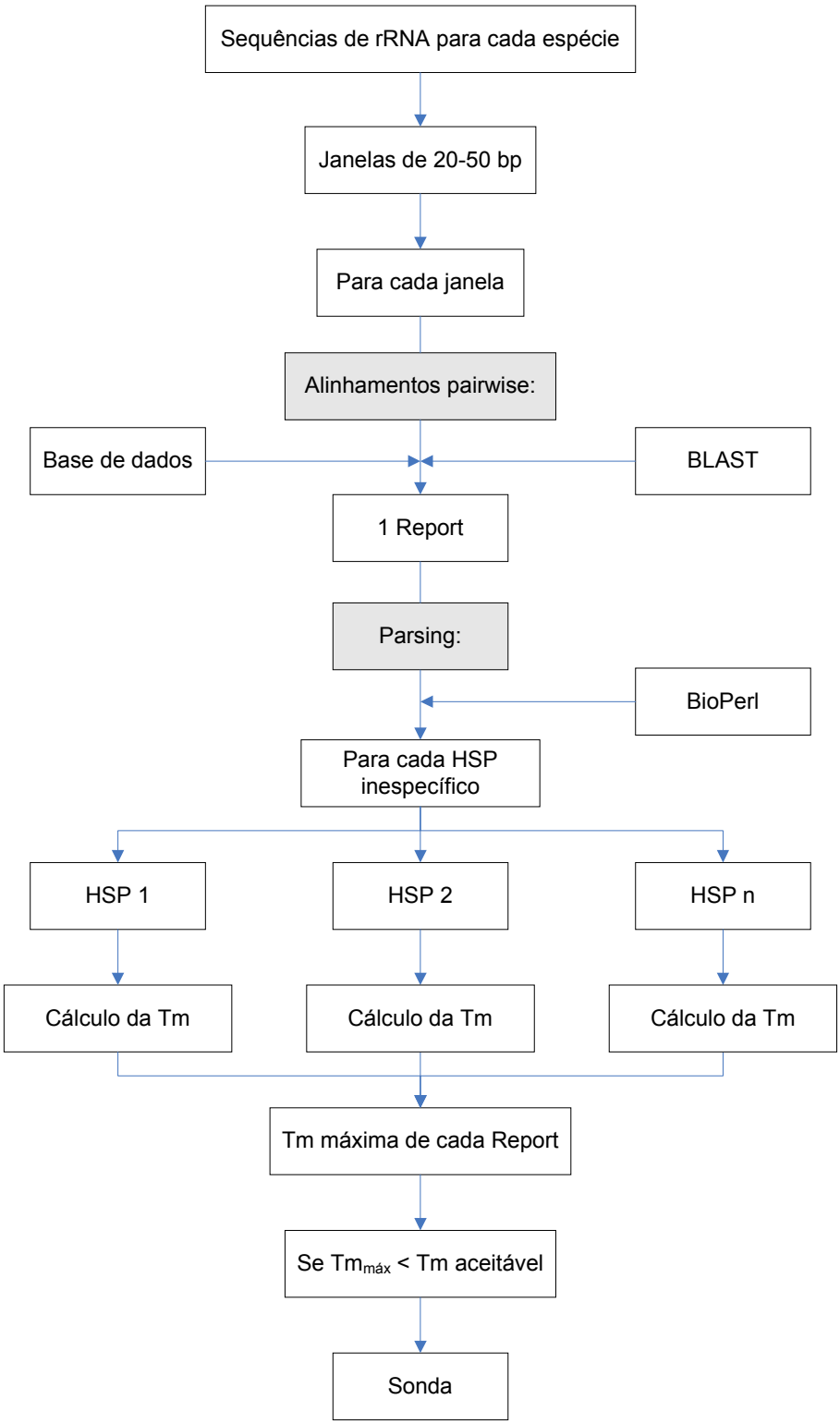

Figura 31 – Fluxograma do modo de funcionamento do *script* jblast3d.pl. Este programa realiza a pesquisa sistemática das janelas na base de dados (Alinhamentos *pairwise*) realizando de seguida o *parsing* e a ordenação dos relatórios com base na Tm dos alinhamentos dos seus HSP.

O processo de cálculo da Tm de cada alinhamento está apresentado na Figura 32. É utilizado o método básico porque é o mais apropriado para sequências pequenas, como são geralmente as dos *matches*.

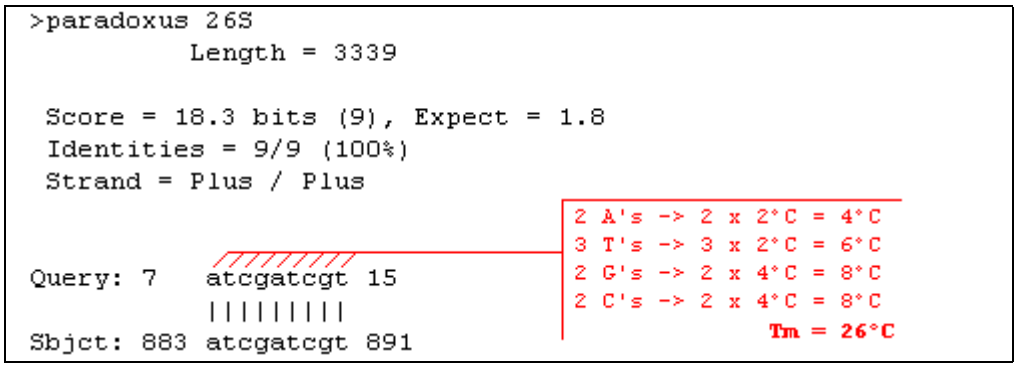

Figura 32 – Pormenor de um relatório de BLAST com determinação da Tm (integrada no *script* jblast4b.pl) de um dos seus HSP. Este alinhamento, correspondente a uma hibridação inespecífica, tem uma temperatura de fusão de 26 ºC. O cálculo é realizado pelo método básico, que adiciona, à Tm, um incremento de 2 ºC por cada par A-T e 4 ºC por cada par G-C.

Para cada relatório, apenas é tida em conta a mais elevada temperatura encontrada num dos seus *Hits* inespecíficos. É com base neste valor que, finalmente, os relatórios são alinhados.

Os resultados finais (ordem dos relatórios segundo as Tm), foram apresentados num ficheiro HTML com hiperligações para todos os dados envolvidos (tal como no *script* anterior). As sequências correspondentes às melhores sondas identificadoras de uma espécie foram aquelas que possuem os menores valores de Tm dos *Hits* inespecíficos - hibridação inespecífica termodinamicamente mais instável.

Numa aproximação teórica, considera-se que a ligação entre uma sonda para espécie A e uma sequência de DNA de espécie B não se estabelece (devido à sua instabilidade), caso a sua temperatura de fusão seja inferior a 30ºC. No entanto, este valor não é definitivo porque a Tm máxima aceitável dos *matches* inespecíficos está relacionada com as condições de hibridação do chip no protocolo laboratorial.

O programa desenvolvido nesta secção (tal como nas outras) ultrapassou essa incerteza. Ao fazer a classificação ordenada das janelas (hipotéticas sondas), deixou ao utilizador a capacidade de seleccionar as sondas mais apropriadas às condições de hibridação que irá utilizar. Este poderá, por exemplo, escolher apenas a sonda melhor cotada (com o menor Tm dos *matches* inespecíficos) ou então alargar o critério e

seleccionar um maior número de sondas – algumas com uma maior probabilidade de estabelecerem hibridação inespecífica.

Foi implementada, num *script* de PERL, a determinação da Tm a partir do método de *Nearest-Neighbour* (Anexo 9). Este método requer o conhecimento das condições laboratoriais em que são realizadas as reacções de hibridação, nomeadamente a concentração das cadeias de oligonucleótidos e a concentração de sais na solução. Esta rotina de cálculo termodinâmico da temperatura de fusão poderá ser facilmente adaptada aos *scripts* (como por exemplo o jblast3d.pl) em substituição do método básico.

4.3.1. Interface gráfica HTML/CGI.

Para maior facilidade de utilização, foi criada uma interface gráfica em HTML para enviar (de acordo com as normas do *Common Gateway Interface*, CGI) todos os dados necessários ao *script* anterior.

Os dados - localização do(s) ficheiro(s) com a(s) sequência(s) e indicação do tamanho das janelas a pesquisar - são introduzidos no formulário de uma página HTML sendo de seguida submetidos (pelo método POST) para o *script* através do Servidor de HTTP Apache.

O código de programação foi ligeiramente modificado (Anexo 10) para obedecer às normas CGI, no que se refere ao uso das variáveis de ambiente e ao *input* dos argumentos para o STDIN (*standard input*) do *script*.

4.4. Sondas pequenas com base em *mismatches* nas posições centrais.

Este método baseia-se na utilização de sondas com sequências específicas mais pequenas - 15 nucleótidos. Estas sondas possuirão, também, uma cauda de 15 Timinas (ligada à superfície do chip pelo grupo 5'amino) para optimizar a acessibilidade e hibridação às sequências da amostra a testar (Figura 33).

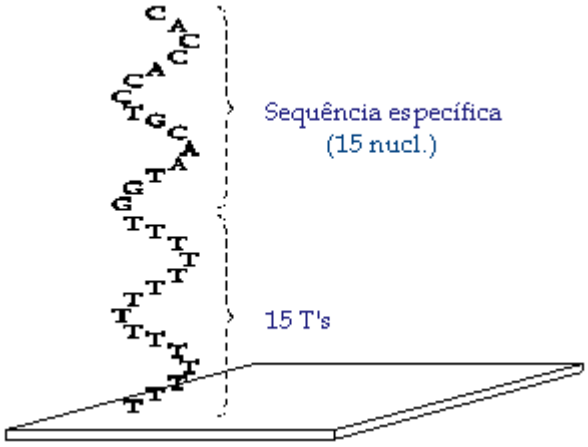

Figura 33 – Representação de uma sonda pequena acoplada à superfície de um chip. A cauda de 15 T's deixa a sequência de quinze nucleótidos específicos acessível para hibridar com uma sequência complementar da amostra.

Este método é utilizado no rastreio de SNP's [126] e proporciona uma alta especificidade entre sonda e sequência de DNA da espécie para a qual é dirigida (são complementares em toda a sua extensão). Uma diferença num dos nucleótidos do meio da sonda (N7) é suficiente para não haver hibridação devido à forte destabilização da dupla hélice.

Deste modo, as aproximações para seleccionar estas sondas específicas tiram partido deste facto, obedecendo às seguintes premissas:

1- A sonda tem que ser totalmente complementar à sequência de DNA da espécie para a qual é dirigida.
2- A sonda terá que possuir nucleótidos diferentes das sequências das outras espécies pelo menos nas posições centrais (podendo o resto ser complementar ou não).

4.4.1. Identificação de sondas a partir de pesquisas sistemáticas na base de dados.

O programa jsearch9.pl (Anexo 11) tem uma lógica de procedimentos semelhante aos anteriores, mas em vez de recorrer ao BLAST como algoritmo de pesquisa, utiliza um algoritmo diferente (Figura 34). Cada janela, é procurada na base de dados (que contém a sequências dos genes das várias espécies bem como as suas complementares inversas) e o número de ocorrências não específicas é contado.

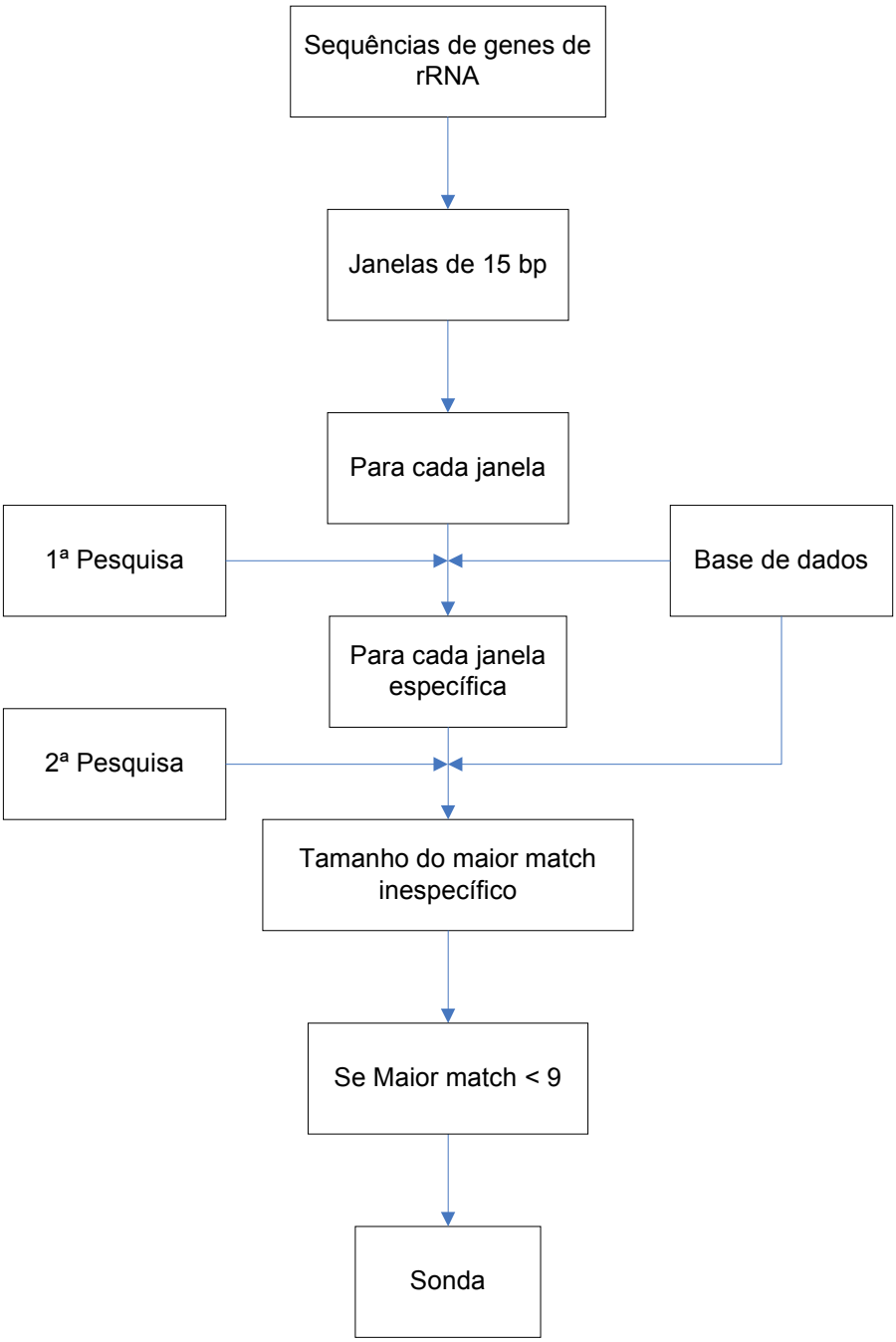

Figura 34 – Fluxograma do modo de funcionamento do *script* jsearch9.pl. Este programa realiza duas pesquisas sequenciais. Na primeira tenta encontrar janelas que só existam numa espécie. Na segunda pesquisa, determina, para essas janelas, o seu maior *match*.

Quando a janela era específica (ocorrências noutras espécies igual a zero, obedecendo à 1ª premissa) iniciou-se uma segunda pesquisa para encontrar o maior *match* (inespecífico, contíguo, sem *gaps* nem nucleótidos diferentes) da janela na base de dados. Para o encontrar, fizeram-se procuras de subjanelas da janela específica na base de dados. Quando o maior *match* encontrado foi inferior a metade do tamanho da janela, isso indicou

que a janela obedeceu à segunda premissa; ou seja, não se encontrou outra espécie na base de dados com quem formar uma dupla hélice hibridada. Para uma sonda de 15 bp, o ideal seria que esse maior *match* fosse igual ou inferior a 7 nucleótidos, mas aceitam-se valores de 8 ou mesmo 9, para obviar bloqueios no *design* das sondas.

Teoricamente estas pesquisas poderiam ser realizadas pelo BLAST, mas este novo algoritmo tornou a produção de resultados mais rápida (não ocorre produção de relatórios nem o seu *parsing*). Os resultados finais foram colocados directamente num ficheiro, apresentando, para cada gene (exemplo: SSU *Candida albicans*), a sequência das sondas encontradas.

Numa tentativa de melhor caracterizar as janelas específicas, foram escritos outros *scripts* com algumas modificações significativas em relação a este. A introdução de uma terceira e quarta pesquisas serviram para determinar o comportamento dos nucleótidos das janelas específicas (e os da base de dados que com eles ficam "alinhados") fora no maior *match*. Por exemplo, quando este foi igual a 8 (Figura 35), na 3ª pesquisa determinou-se se os restantes formariam um *match* contíguo, num máximo de 6 nucleótidos (sonda de 15 - 8 do maior *match* - 1 *mismatch*), e na 4ª contaram-se todos os nucleótidos coincidentes com os da base de dados (nas zonas que poderiam hibridar com a sonda).

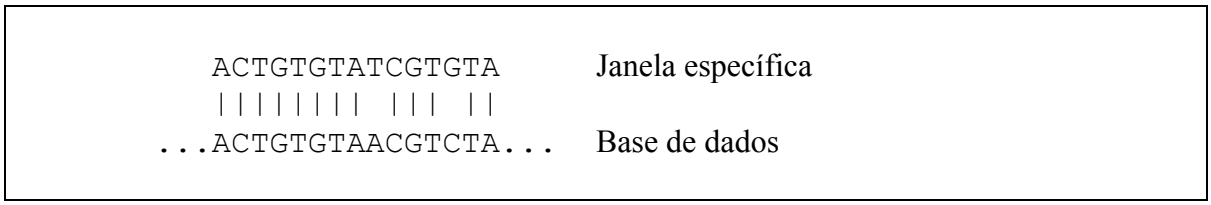

Figura 35 – Exemplo de contagem de nucleótidos no *script* jsearch9.pl e *scripts* semelhantes. A 1ª pesquisa identificou esta janela como sendo específica. Na 2ª pesquisa detectou-se o maior *match* da janela com a base de dados (8 nucleótidos; ACTGTGTA) - o processamento de jsearch9.pl termina com a indicação de que esta janela pode funcionar como sonda. A 3ª pesquisa determinou que o 2º maior *match* tem 3 nucleótidos (CGT). A 4ª conta 5 nucleótidos (CGT e TA) como sendo coincidentes fora do maior *match*.

Embora o objectivo destas modificações seja aprofundar a caracterização dos *matches* da janela na base de dados, as classificações resultantes tornam-se algo complexas e de avaliação directa difícil.

4.4.2. Identificação de sondas a partir de alinhamentos múltiplos.

O segundo método para seleccionar este tipo de sondas partiu de um princípio radicalmente diferente dos referidos anteriormente. Baseou-se na utilização de alinhamentos múltiplos para comparar as sequências dos genes das várias espécies. O algoritmo Clustal introduziu os *gaps* necessários para que os padrões de nucleótidos ficassem alinhados em todas as sequências e colocou este alinhamento num ficheiro. O *script* compmultalign4.pl (Anexo 12), representado na Figura 36 fez o *parsing* deste ficheiro e inferiu as possíveis sondas.

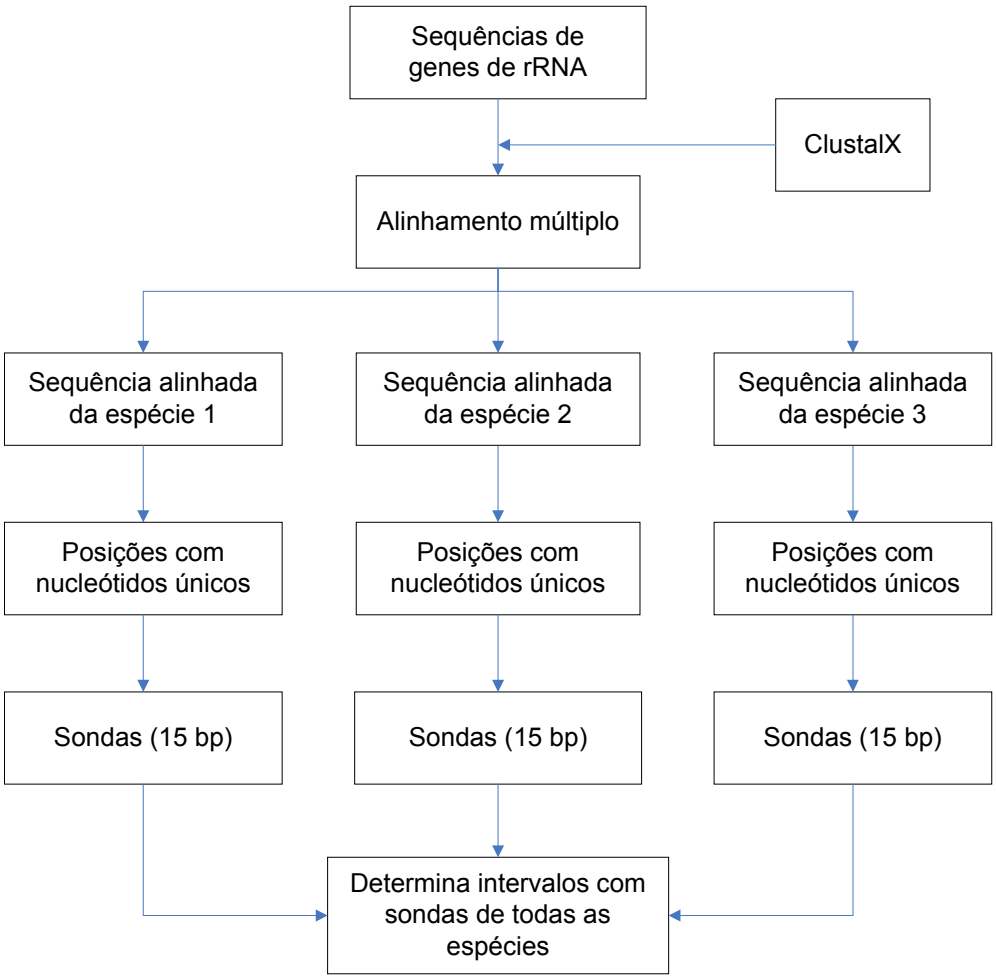

Figura 36 – Fluxograma do modo de selecção de sondas pequenas recorrendo a alinhamentos múltiplos. Após o algoritmo Clustal realizar o alinhamento de todas as sequências, o *script* compmultalign4.pl procurou as posições com nucleótidos únicos de cada uma em relação às outras. A sequência de 15 nucleótidos com posição central única será uma sonda. Finalmente, determinou intervalos do alinhamento em que todas as espécies possuíssem pelo menos uma sonda.

Para cada sequência do alinhamento, este programa detectou as posições em que possui um nucleótido diferente de todas as outras sequências (Figura 37). Estas posições

com nucleótidos únicos permitiram desenhar sondas de 15 nucleótidos obedecendo às duas premissas descritas da secção 4.4. Pressupõe-se que são sondas específicas por causa do nucleótido único (e porque será praticamente impossível encontrar sequências iguais de 15 nucleótidos noutras zonas dos genomas fora do alinhamento). Se a sonda fosse desenhada de forma a conter o nucleótido único na posição 8, a segunda premissa seria igualmente obedecida.

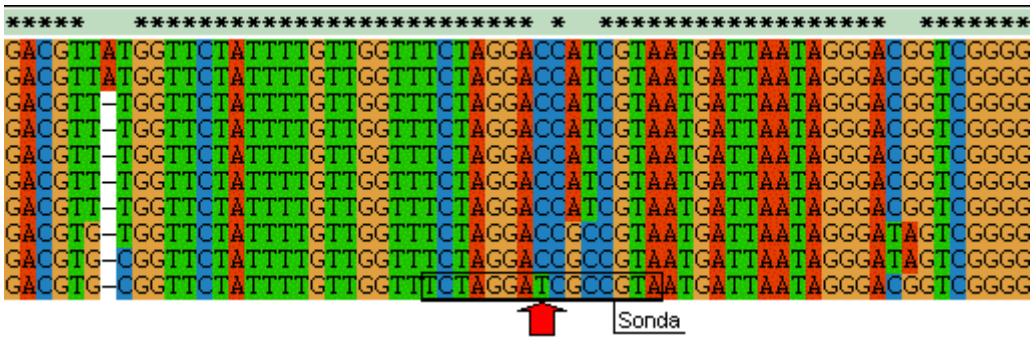

Figura 37 – Pormenor de um alinhamento múltiplo de 10 sequências de espécies diferentes realizado pelo Clustalx Na posição assinalada com uma seta existe um nucleótido único para a última sequência. Com base nesta diferença é possível desenhar uma sonda específica de 15 nucleótidos para essa espécie.

Após a determinação de todos as posições com nucleótidos únicos do alinhamento, o *script* calculou os vários intervalos do alinhamento em que todas as sequências representadas possuíam pelo menos um desses valores. No final foi determinado o mais pequeno destes intervalos. Este valor foi importante para indicar as zonas da amostra a ser amplificadas por PCR (ver secção 6.).

5. Validação das sondas escolhidas localmente na base de dados do NCBI.

Após se ter seleccionado conjuntos de sondas específicas através de sistemas e de bases de dados locais (com um número de espécies limitado), foi necessário validar estas sequências contra a base de dados "nr" ("não-redundante") instalada nos servidores Web do NCBI. Esta inclui todas as sequências do GenBank, Refseq, EMBL, DDBJ e PDB (não inclui sequências EST, STS, GSS ou HTGS das fases 0, 1 e 2). Ao contrário do que o seu nome indica, esta base de dados já inclui redundâncias. Foi apenas pesquisada a subsecção *Fungi* desta base de dados, por ser aquela que apenas inclui sequências de espécies de Fungos.

A pesquisa foi realizada recorrendo ao programa BLAST igualmente instalado nos servidores do NCBI.

A comunicação entre o *script* (a correr na máquina do utilizador) e os servidores do NCBI foi realizada recorrendo à interface URL-API. Esta interface permitiu fazer pedidos *HTTP-encoded* directamente aos servidores através de um programa *cgi-bin*, alojado em: http://www.ncbi.nlm.nih.gov/blast/Blast.cgi. Neste sistema, a pesquisa completa das sequências foi alcançada em dois passos: a submissão da sequência e a obtenção dos resultados sob a forma de relatórios.

5.1. Submissão das sequências ao BLAST.

Para submeter automaticamente as pesquisas no BLAST foi desenvolvido o *script* jqblast2.pl (Anexo 13) representado na Figura 38.

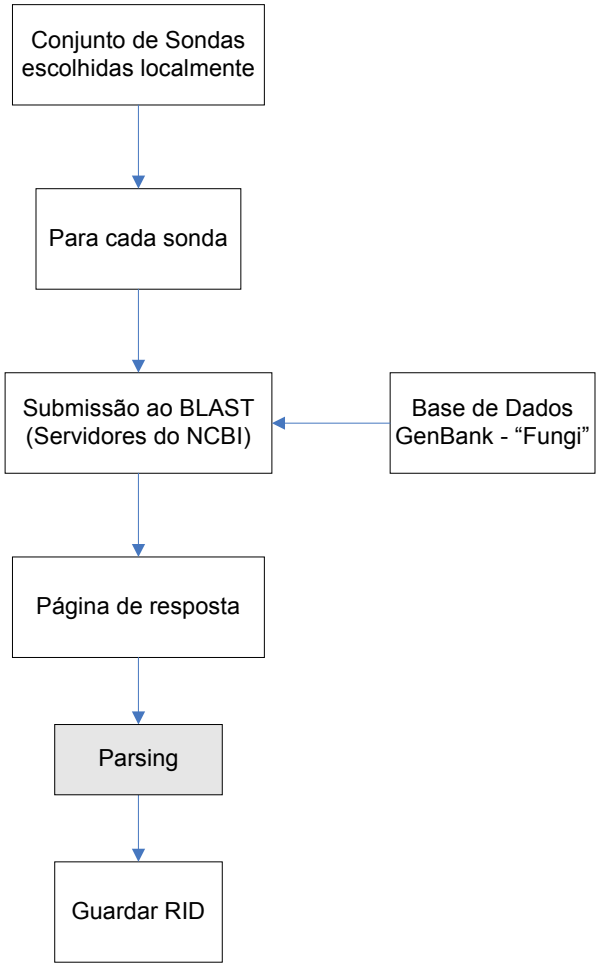

Figura 38 – Fluxograma do modo de submissão das sequências das sondas ao BLAST alojado nos servidores do NCBI. Cada sonda é pesquisada individualmente na subsecção *Fungi* da base de dados "nr", sendo guardado o *Request* ID resultante que é apresentado na página de resposta.

O *input* deste programa foi um ficheiro de texto que apresenta, para cada espécie, as sequências das sondas seleccionadas localmente. Para cada uma é realizado um pedido (pelo método POST) ao servidores do NCBI, com um determinado conjunto de parâmetros (Tabela 2). Destes, um é o comando "CMD=Put" que determina que o pedido tenha como objectivo a submissão de uma sequência ao BLAST. A sequência da sonda, em formato FASTA, é enviada associada ao parâmetro "QUERY" (p.ex.: "QUERY=ACTTGCTTTGGCGGT").

Tabela 2 – Lista e descrição de cada par "parâmetro=valor" essenciais usados nos pedidos ao programa BLAST.cgi (instalado nos servidores do NCBI) para a submissão das sequências das sondas (seleccionadas localmente) para pesquisa.

<Parâmetro>=<Valor>	**Descrição**
CMD=Put	Comando que determina a submissão de sequências para o BLAST
PROGRAM=Blastn	Nome do programa a utilizar - BLASTn
DATABASE=nr	Utilizar base de dados "nr" na pesquisa
ENTREZ_QUERY=Fungi+[ORGN]	Subsecção *Fungi* da base de dados "nr"
QUERY=Sequência	Sequência de nucleótidos de cada possível sonda
FILTER=L	Filtrar as sequências com pouca complexidade (ex: AAAAAA)

Após enviado o pedido, o *script* guarda a resposta, sob a forma de uma página HTML. Esta página contém o código do *Request Identifier* (RID) - o identificador único de cada pesquisa nos servidores do BLAST. A partir do RID (por exemplo: 1092414774-13488-181709615095.BLASTQ4), é possível obter o relatório da pesquisa, assim que ele fique pronto. Devido à sobrecarga dos seus servidores, com pedidos de BLAST originários de todo o mundo, uma pesquisa pode demorar entre alguns segundos até vários minutos a ser finalizada. Nas situações em que vários pedidos são realizados a partir do mesmo endereço IP, como é o caso deste ensaio, cada submissão sucessiva incorre numa penalização de tempo adicional. Deste modo, não seria viável proceder à extracção do relatório logo de seguida, pelo que após a submissão de cada sonda realiza-se o *parsing* da página HTML de resposta e guarda-se o RID localmente.

No final deste processo obtém-se apenas um ficheiro de texto com a lista dos códigos RID, associados às sequências pesquisadas que lhes deram origem. De realçar que cada RID tem uma validade limitada de apenas 24 horas, pelo que após esse período, caso não tenham sido extraídos os relatórios, terá que ser realizada nova submissão de sequências.

5.2. Obtenção e parsing dos relatórios.

Para extrair os relatórios dos servidores foi desenvolvido o *script* jqblast2b.pl (Anexo 14). Este *script*, descrito na Figura 39, recebeu os RID do ficheiro de texto, um de cada vez, e fez novos pedidos (também pelo método POST) ao programa *cgi-bin* alojado em: http://www.ncbi.nlm.nih.gov/blast/Blast.cgi. Entre cada pedido ocorreu uma espera de 60 segundos para não sobrecarregar os servidores do NCBI.

Desta vez, o comando que determina o tipo de acção foi "CMD=Get", para a extracção dos relatórios. Os restantes parâmetros foram "AUTO_FORMAT=Fullauto", para apresentar imediatamente os resultados numa página e "RID=<RID>" em que <RID> é o código do *Request Identifier*.

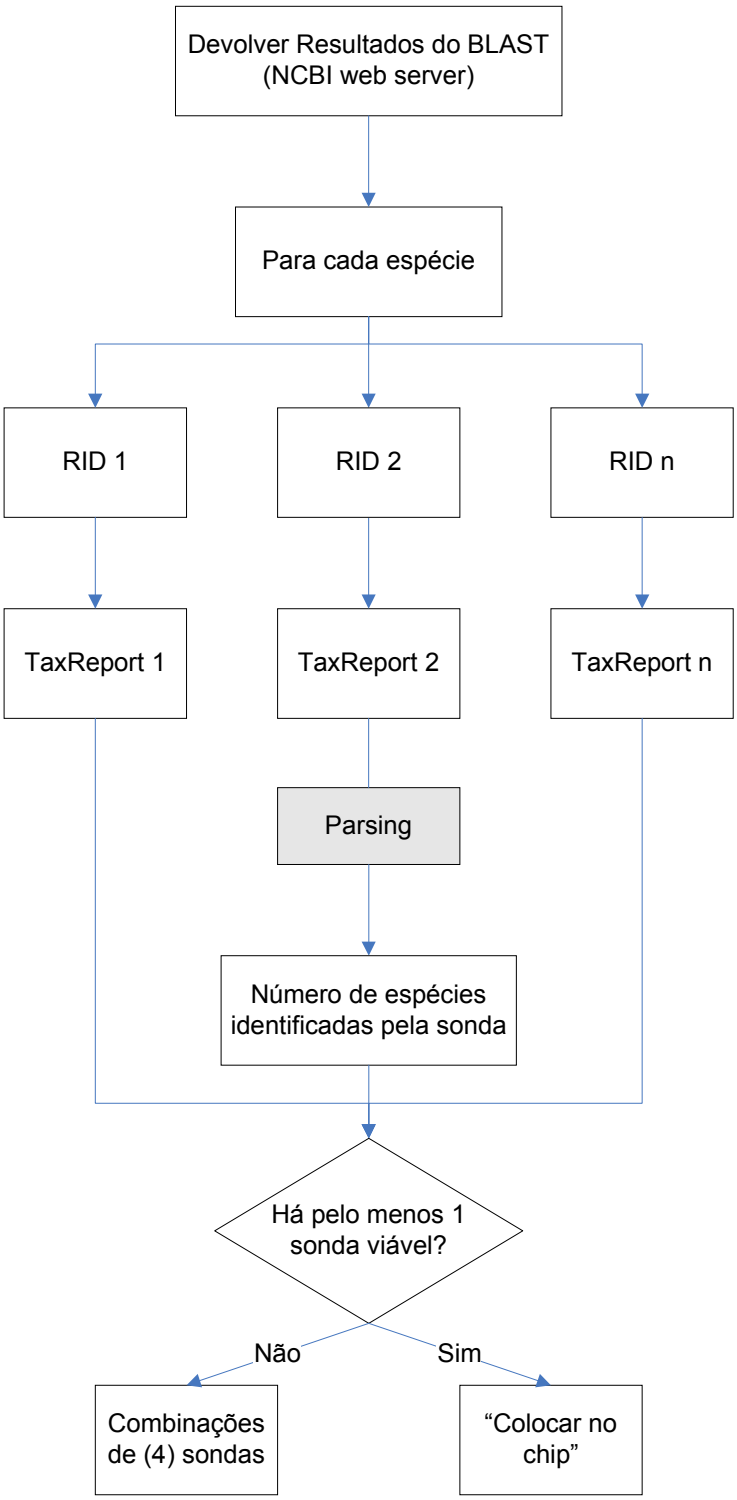

Figura 39 – Fluxograma do modo de obtenção e *parsing* dos relatórios de BLAST. A partir do RID de cada pesquisa foi possível obter os relatórios dos servidores do NCBI. Em cada *TaxReport*, foi indicado o número de espécies identificadas pela sonda que lhe deu origem. Se, para uma espécie, houverem sonda viáveis, utilizam-se no chip. Caso contrário, recorre-se a combinações de sondas para a identificar.

Para cada RID, são extraídos dois ficheiros diferentes resultantes da mesma pesquisa. O relatório tradicional e o *TaxReport* (adicionando o parâmetro

"FORMAT_OBJECT=TaxBlast" num segundo pedido), um tipo de relatório que apresenta os resultados com base na classificação taxonómica formal das espécies referidas nos resultados da pesquisa. Este ficheiro, embora apresente menor quantidade de informação referente aos *matchs* e alinhamentos, não apresenta redundâncias na forma de se referir à mesma espécie. Enquanto que num relatório os resultados estão agrupados de acordo com as entradas das sequências das bases de dados (podendo existir várias entradas para a mesma espécie), num *TaxReport*, todos os *matchs* da mesma espécie estão agrupados, facilitando a extracção da informação.

Se por algum motivo qualquer dos relatórios não estava pronto quando foi guardado localmente – ou porque a pesquisa ainda não se realizou ou por dificuldades de comunicação –, fizeram-se sucessivos novos pedidos aos servidores até que se recebeu o relatório completo. Isto foi possível determinar automaticamente porque existe uma linha no início de cada relatório que indica a sua condição: "status=WAITING ou "Status=READY".

À medida que cada *TaxReport* foi recebido, fez-se imediatamente o seu *parsing* (os relatórios normais não foram analisados sendo guardados apenas como referência), determinando-se o nome e o número de espécies em que foi encontrada a sequência da sonda em questão. Os resultados finais foram apresentados num ficheiro HTML com indicação da sequência da sonda, as hiperligações para relatório e *TaxReport* e o número de espécies contado anteriormente. Se a sequência existia apenas numa espécie, a linha apresentou a cor verde, para uma visualização mais fácil das sondas específicas; caso contrário apresentou a cor vermelha.

5.2.1. Combinações de sondas.

Para aquelas espécies (ou genes de espécies) para os quais não foi possível encontrar sondas específicas após a pesquisa na base de dados do NCBI, tentou-se encontrar combinações específicas de sequências não específicas. O *script* j14b.pl (Anexo 15) tem como *input* um ficheiro de texto (produzido por jqblast2b.pl) com as sequências (de sondas seleccionadas localmente mas não validadas no NCBI) dessas espécies e as listas dos nomes das espécies em que elas existem. O *script* analisa todas as combinações de sequências 4 a 4 e determina as que são específicas, isto é, aquelas em que apenas uma das espécies (aquela para qual a sonda é dirigida) aparece nas 4 sequências.

Este processo parte do princípio de que apenas uma espécie conterá todas as sequências. Se houver outra espécie nas mesmas condições, será logicamente impossível encontrar combinações específicas.

6. Selecção e caracterização dos *primers* de PCR.

Para desenhar os pares de *primers* para o processo de PCR foi necessário, em primeiro lugar, determinar as secções de sequências da amostra a amplificar. A partir destas, tentou encontrar-se zonas adjacentes que sejam homólogas entre todas as espécies e diferentes das do genoma humano – recorrendo a alinhamentos múltiplos. Os *primers* de PCR foram seleccionados dentro dessas zonas.

Foi escrito o *script* primer3.pl (Anexo 16) para fazer essa selecção. O seu funcionamento, representado na Figura 40, é relativamente semelhante a *scripts* anteriores de selecção de sondas. Escolhe janelas dentro das zonas homólogas e de seguida caracteriza-as segundo alguns parâmetros.

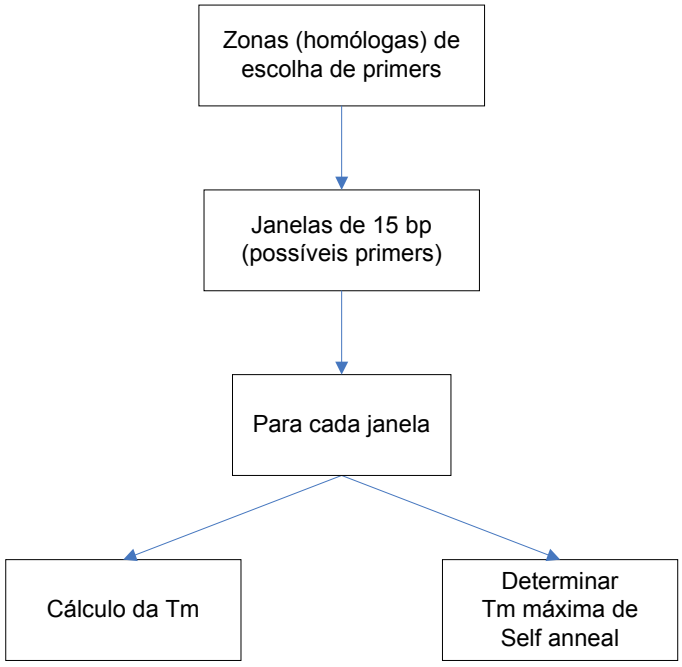

Figura 40 – Fluxograma do processo de selecção e caracterização dos *primers* de PCR. O programa selecciona janelas de nucleótidos nas zonas homólogas adjacentes à sequência a amplificar e determina a sua Tm e a Tm máxima de *self anneal*.

Para cada janela calcula a temperatura de hibridação da sua sequência (segundo o método básico). Especial atenção foi dada à Tm para que esta fosse idêntica para todos os

primers, de modo a garantir que numa única reacção de PCR fosse possível amplificar várias sondas.

De seguida, o *script* analisou a probabilidade de ocorrer *self-anneal*, ou seja, a capacidade do *primer* hibridar com ele próprio, que não é desejado devido a bloquear a amplificação na reacção de PCR. Essa análise foi realizada calculando (os pares de) subsecções da janela que poderiam hibridar umas com as outras. Finalmente, determinou-se o par com a maior Tm (método básico). As janelas com menor Tm máxima de *self-anneal* são as que têm menor probabilidade de auto-hibridar. Verificou-se que, dado que os *primers* são sequências muito pequenas, o *self-anneal* é improvável, a não ser nos casos extremos em que metade (ou quase) do *primer* hibride com a outra metade.

Os resultados da caracterização de cada janela foram enviados para um ficheiro resumo, a partir do qual foi feita a selecção dos *primers* que melhor se adaptam às condições laboratoriais da amplificação das sequências por PCR.

Este programa pode igualmente ser usado para caracterizar a probabilidade de *self-anneal* das sondas, dado que é um fenómeno que pode influenciar negativamente e em grande medida a eficiência de hibridação do chip.

Capítulo III: Resultados

Para além de desenvolver os sistemas bioinformáticos necessários para o desenho de *chips* de DNA de diagnóstico molecular, este projecto também teve como objectivo a sua aplicação prática no desenho de alguns *chips*. Neste capítulo, são apresentados os resultados produzidos através da execução de alguns desses sistemas, que permitirão o desenho de dois *chips* de DNA de diagnóstico. Nesse sentido, são identificadas as sequências das sondas específicas para cada uma das espécies consideradas. No primeiro *chip*, foram seleccionadas sondas de 50 nucleótidos para identificar as espécies *C. albicans* e *S. cerevisiae*. No segundo *chip*, constituído por sondas de 15 nucleótidos específicos, foram incluídas, para além das duas espécies anteriores, *A. fumigatus*, *C. glabrata*, *C. tropicalis*, *C. neoformans*, *S. bayanus*, *S. mikatae*, *S. paradoxus* e *S. pombe*. Por último, são indicados os *primers* seleccionados para a amplificação, por PCR, das sequências de rRNA que contêm os alvos para as sondas, nas 10 espécies.

1. Montagem das sequências completas do rDNA.

As sequências do rDNA foram extraídas das bases de dados de sequências de nucleótidos online. No Anexo 1, podem ser encontrados os códigos de acesso das entradas, referentes a essas bases de dados, que continham as sequências analisadas neste projecto. A montagem das sequências completas do rDNA, foi realizada tal como está descrito no diagrama da figura 24. As sequências parciais extraídas das bases de dados foram alinhadas e unidas numa única sequência total, com remoção das regiões sobrepostas entre sequências adjacentes. A sequência completa do rDNA de cada espécie ficou constituída pelas sequências contíguas dos genes e regiões não codificantes 18S, ITS1, 5.8S, ITS2 e 28S.

2. Desenho de um *FunChip* com sondas de 50 nucleótidos.

O programa jblast3d.pl (descrito na secção 4.3 da metodologia) foi aplicado ao desenho de um mini-*chip* para identificação das espécies *C. albicans* e *S. cerevisiae* em amostras laboratoriais. As regiões de interesse cujas sequências foram utilizadas para realizar a análise, foram os *spacers* ITS1 e ITS2 das duas espécies. Janelas destas

sequências de DNA foram pesquisadas sobre a base de dados com o algoritmo BLAST e para cada uma foi produzido um relatório contendo os resultados dessa pesquisa.

Foi realizada a análise de cada relatório, calculando-se o valor de Tm máximo dos *hits* inespecíficos, ou seja, aquelas secções da sequência da janela pesquisada que também se encontram presentes em sequências de outras espécies da base de dados. As janelas seleccionadas como sondas são aquelas que possuem os menores valores desta Tm, pois não permitem a formação de uma dupla cadeia estável em situações de hibridação cruzada.

As sondas escolhidas, com uma extensão de 50 nucleótidos estão representadas na Tabela 3. Para cada espécie são indicadas as sequências de 8 possíveis sondas, 4 de cada ITS. Para cada sonda é igualmente indicada a Temperatura de fusão (Tm), o maior valor de Tm encontrado entre os seus *hits* inespecíficos ($Tm_{máx\_inesp.}$) e as suas coordenadas de localização na sequência total do ITS em questão.

Tabela 3 – Sondas de 50 nucleótidos seleccionadas nas regiões ITS1 e ITS2 para as espécies *C. albicans* e *S. cerevisiae*. São indicadas as coordenadas do 1º nucleótido, a Tm e a Tm máxima inespecífica.

***C. albicans* ITS1**			
Coord.	$Tm_{máx\_inesp.}$ (ºC)	Tm (ºC)	Sequência
88	22	128	TTACAACCAATTTTTTATCAACTTGTCACACCAGATTATTACTAATAGTC
8	28	140	GCTTAATTGCACCACATGTGTTTTTCTTTGAAACAAACTTGCTTTGGCGG
16	28	148	GCACCACATGTGTTTTTCTTTGAAACAAACTTGCTTTGGCGGTGGGCCCA
60	28	148	GGCCCAGCCTGCCGCCAGAGGTCTAAACTTACAACCAATTTTTTATCAAC
***C. albicans* ITS2**			
Coord.	$Tm_{máx\_inesp.}$ (ºC)	Tm (ºC)	Sequência
59	28	146	GTAGTGGTAAGGCGGGATCGCTTTGACAATGGCTTAGGTCTAACCAAAAA
73	28	148	GGATCGCTTTGACAATGGCTTAGGTCTAACCAAAAACATTGCTTGCGGCG
87	28	150	ATGGCTTAGGTCTAACCAAAAACATTGCTTGCGGCGGTAACGTCCACCAC
101	28	146	ACCAAAAACATTGCTTGCGGCGGTAACGTCCACCACGTATATCTTCAAAC
***S. cerevisiae* ITS1**			
Coord.	$Tm_{máx\_inesp.}$ (ºC)	Tm (ºC)	Sequência
273	20	124	CAGAGGTAACAAACACAAACAATTTTATCTATTCATTAAATTTTTGTCAA
94	24	156	CCGGGCCTGCGCTTAAGTGCGCGGTCTTGCTAGGCTTGTAAGTTTCTTTC
122	24	140	GCTAGGCTTGTAAGTTTCTTTCTTGCTATTCCAAACGGTGAGAGATTTCT
312	24	118	ATTTTTGTCAAAAACAAGAATTTTCGTAACTGGAAATTTTAAAATATTAA
***S. cerevisiae* ITS2**			
Coord.	$Tm_{máx\_inesp.}$ (ºC)	Tm (ºC)	Sequência
62	22	140	GCCTTTTCATTGGATGTTTTTTTTCCAAAGAGAGGTTTCTCTGCGTGCTT
10	24	136	ACATTCTGTTTGGTAGTGAGTGATACTCTTTGGAGTTAACTTGAAATTGC
152	24	136	TGCGGCTAATCTTTTTTTATACTGAGCGTATTGGAACGTTATCGATAAGA
108	26	146	GCTTGAGGTATAATGCAAGTACGGTCGTTTTAGGTTTTACCAACTGCGGC

As sondas estão indicadas na orientação 5'-3'.

Pela análise da Tabela 3, é possível verificar que todas as sondas seleccionadas apresentam valores de Tm inespecífica máxima inferiores a 28 ºC. As sondas do ITS2 de *C. albicans* são as que possuem, em geral, valores mais altos. Por seu lado, no ITS1 de *S. cerevisiae* foi possível encontrar pelo menos 4 sondas com valores inferiores a 24 ºC.

Estas sondas foram seleccionadas igualmente no intuito de se obter a maior representatividade possível da sequência do ITS em questão. Para isso, foi evitada a selecção de janelas próximas ou mesmo adjacentes (por exemplo, janela1 e janela2 da Figura 27). A maior sobreposição de duas sondas é observada entre a segunda e terceira sondas do ITS1 de *C. albicans*, com início nas posições 8 e 16 desta estrutura respectivamente, que apresentam sequências iguais em 42 nucleótidos.

A análise das sondas com o programa primers3.pl, permitiu concluir que não é possível ocorrer formação significativa de estrutura secundária "no interior" de cada cadeia. A maior probabilidade de ocorrência deste fenómeno verifica-se na última sonda de *S. cerevisiae* ITS1, em que a hibridação entre os fragmentos, inversamente complementares, TTTTTG e CAAAAA (nas posições 1-6 e 8-13 da sonda, respectivamente) poderia originar a formação de um gancho (*hairpin*) com um T (posição 7) protuberante.

A partir de sondas de oligonucleótidos com estas sequências, é possível desenhar um *chip* capaz de distinguir *C. albicans* de *S. cerevisiae* em amostras laboratoriais.

3. Desenho de um *FunChip* com sondas de 15 nucleótidos específicos.

As espécies escolhidas para testar o sistema de desenho de *chips* constituído pelos programas jsearch9.pl, compmultalign4.pl, jqblast2.pl, jqblast2b.pl e j14b.pl (descrito nas secções 4.4 e 5 da metodologia) foram os fungos *Aspergillus fumigatus*, *Candida albicans*, *Candida glabrata*, *Candida tropicalis*, *Cryptococcus neoformans*, *Saccharomyces bayanus*, *Saccharomyces cerevisiae*, *Saccharomyces mikatae*, *Saccharomyces paradoxus* e *Schizosaccharomyces pombe*.

Este *chip* utilizará sondas de oligonucleótidos 30-mer, constituídas por 15 timinas (T) não específicas e por uma sequência específica de 15 nucleótidos. Nesta secção convenciona-se que a designação "sonda" se refere apenas a esta segunda parte, uma vez que a cauda de T's serve como espaçador entre a superfície sólida e a sequência específica.

Os critérios de especificidade aplicados na selecção destas sondas determinam que a sequência da sonda tem que ser totalmente complementar apenas com a sequência de DNA da espécie para a qual é direccionada e tem que possuir nucleótidos diferentes em relação às sequências das outras espécies pelo menos nas suas posições centrais. Se uma sonda obedecer as estes critérios, irá hibridar e formar uma dupla hélice termodinamicamente estável unicamente com a espécie para a qual foi desenhada. O resultado desta reacção será um sinal fluorescente observado no ponto de hibridação correspondente.

Em primeiro lugar, foram aplicados os dois sistemas locais, jsearch9.pl e compmultalign4.pl (descritos nas secções 4.4.1 e 4.4.2 da metodologia, respectivamente), para a selecção de sondas de 15 nucleótidos nos genes 18S, ITS1, ITS2 e 28S das 10 espécies em estudo. O primeiro programa determinou para cada janela específica (sequência de nucleótidos existente numa só espécie da base de dados) o tamanho do maior *match* inespecífico contíguo de nucleótidos. Se este tinha uma extensão inferior ou igual a 9, a janela pôde ser seleccionada como sonda específica. O programa compmultalign4.pl baseou-se na análise de todas as posições dos alinhamentos múltiplos entre as sequências do rDNA das 10 espécies. Para cada posição em que foi identificado um nucleótido único (que apenas se encontra na sequência de uma das espécies), foi desenhada uma sonda constituída pelos sete nucleótidos imediatamente anteriores, pelo nucleótido único na posição 8, e pelos sete imediatamente posteriores.

As sequências dos ITSs foram totalmente pesquisadas pelos dois programas, enquanto que as dos genes dos rRNA 18S e 28S apenas foram pesquisados em secções parciais. Estas secções foram determinadas pelo programa compmultalign4.pl que calculou os intervalos dos alinhamentos que incluíam o maior número de sondas para as 10 espécies. Para o gene do rRNA 18S foi extraída a secção do alinhamento (Anexo 2) entre as posições 638 e 828 e para o gene 28S a secção compreendida entre as posições 346 e 681 do alinhamento (Anexo 4). O *script* jsearch9.pl apenas analisou estes intervalos destes genes, pesquisando todas as janelas contra a base de dados constituída pelas sequências de todos os genes do rDNA das 10 espécies bem como as suas complementares inversas.

As sondas seleccionadas pelos dois programas foram reunidas e as sequências redundantes foram eliminadas. Estas sondas permitem a distinção entre as 10 espécies em

estudo, mas poderiam existir, adicionalmente, no genoma de outras espécies não tidas em conta. Portanto, de seguida, procedeu-se à sua validação pesquisando as sondas na base de dados "nr", subsecção *Fungi*, do NCBI, com o *Web*-BLAST. Este processo foi realizado tal como está descrito na secção 5 da metodologia. As sequências candidatas a sondas foram agrupadas num ficheiro de texto que serviu de *input* ao jqblast2.pl. Este submeteu-as como sequências *query* ao BLAST do NCBI e extraiu os RID fornecidos. Posteriormente, o programa jqblast2b.pl recebeu estes identificadores únicos e extraiu os dois tipos de relatórios do BLAST resultantes de cada pesquisa. Realizou, de seguida, o *parsing* dos *TaxReports* e identificou o número de espécies que possuíam as sequências pesquisadas, no mínimo, numa das suas entradas na base de dados. As sequência às quais corresponde apenas uma espécie, foram, portanto, validadas e consideraram-se como sendo sondas específicas.

Os resultados finais estão sumariados na Tabela 4. Quando não foi possível encontrar uma sonda específica, num dos genes, para uma determinada espécie, foi usada a metodologia de análise combinatória realizada pelo programa j14b.pl (descrito na secção 5.2.1). Este programa analisou todas as combinações possíveis de quatro sondas e identificou aquelas em que apenas uma espécie possui a sequência de todas as sondas. Embora sejam constituídas por sondas não específicas, estas combinações foram consideradas como específicas.

Tabela 4 – Resumo dos resultados finais da selecção de sondas de 15 nucleótidos para as 10 espécies, após a aplicação dos sistemas locais e do processo de validação na base de dados universal "nr" do NCBI. O símbolo "√" assinala os casos em que foi encontrada pelo menos uma sonda (ou combinação de 4 sondas) específica no gene e na espécie referidos. O símbolo "×" assinala os casos em que tal não foi conseguido.

	18S		ITS1		ITS2		28S	
	1 sonda	comb. 4	1 sonda	comb. 4	1 sonda	comb. 4	1 sonda	comb. 4
A. fumigatus	×	×	√		×	√	×	√
C. albicans	√		√		√		√	
C. glabrata	×	×	√		√		√	
C. tropicalis	√		√		√		√	
C. neoformans	×	√	√		×	×	×	√
S. bayanus	×		×	×	×		×	×
S. cerevisiae	×		×		×		×	×
S. mikatae	×		√		×		×	
S. paradoxus	×		×		×		×	
S. pombe	√		√		√		√	

Foram encontradas sondas totalmente específicas para todos os genes de *C. albicans*, *C. tropicalis* e *S. pombe*. Na pesquisa das sondas do ITS1 de *C. albicans* foi encontrada uma entrada (com o código de acesso AY342214) com o nome específico de *Candida africana*, que também as incluía. No entanto, nas anotações desta entrada (a única com este nome em todo a base de dados) é referido que *C. africana* é provavelmente um sinónimo de *C. albicans*, logo não foi considerada válida nesta análise. De igual forma, não foi tida em conta uma entrada (AY233752) com o nome específico de Candida sp. CA-SPX-CL096C, presente em todas as pesquisas das sondas de *C. tropicalis*, gene 28S, por ser um código não taxonómico usado apenas numa única experiência.

Para a espécie *A. fumigatus*, apenas não foi possível encontrar sondas específicas no gene 18S, já que todas as sequências pesquisadas foram sequenciadas em muitas outras espécies, particularmente do género *Penicillium*. Devido ao grande número de espécies correspondentes a cada sonda (superior a 60 spp. em algumas), não foi igualmente possível encontrar combinações específicas para este gene.

Para o gene 18S de *C. glabrata* não se determinou nenhuma sonda válida, porque a sequência deste gene de onde foram seleccionadas localmente todas as sondas é exactamente igual na espécie *Kluyveromyces delphensis*. Caso se utilize uma destas sondas no *chip*, será necessário ter em atenção este pormenor e fazer o contraste com as 3 sondas dos outros genes, para distinguir *C. glabrata* de *K. delphensis*. Esta semelhança era previsível, dado que vários estudos realizados têm vindo a demonstrar a grande proximidade filogenética entre estas duas espécies (Kurtzman et al, 2003). As pesquisas das sondas dos dois ITS encontraram ainda *hits* perfeitos em duas entradas (AY589572 e AY589573) sob um nome não validado taxonomicamente, Candida sp. 153M, que não foi tido em conta.

Em relação à levedura *C. neoformans*, foi possível seleccionar sondas específicas com alvo no ITS1, embora a sua determinação tenha sido dificultada pela existência de variantes intraespecíficas, como *C. neoformans var. Neoformans* ou *C. neoformans var. Grubii*. Para os genes 18S e 28S, dado que todas as sequências pesquisadas foram encontradas noutras espécies (embora de forma alternada), a abordagem alternativa permitiu determinar combinações de 4 sondas específicas. As sequências escolhidas no ITS2, revelaram-se demasiado semelhantes entre algumas espécies e nenhuma permitia a identificação unívoca de *C. neoformans*, sendo que determinadas espécies, nomeadamente

a *C. bacillisporus*, foram observadas em todas as sondas, impedindo também a estratégia combinatória.

A identificação discriminatória das quatro espécies do género *Saccharomyces* revelou-se, tal como esperado, de difícil obtenção. As sequências dos genes 18S são quase totalmente conservadas entre *S. cerevisiae*, *S. bayanus*, *S. mikatae* e *S. paradoxus*, com a excepção de um nucleótido de *S. mikatae* na posição 650 do alinhamento (Anexo 2). Esta diferença mínima permitiu desenhar uma sonda de 15-mer com esse nucleótido na posição central. No entanto, a pesquisa na base de dados "nr" devolveu mais duas espécies, *Saccharomyces kudriavzevii e Saccharomyces pastorianus*, com a mesma sequência, inviabilizando a sua colocação no *chip*.

As sequências dos genes 28S de *S. mikatae* e *S. paradoxus* são exactamente iguais o que impossibilitou a selecção local de sondas identificadoras, neste dois casos. Em relação aos genes 28S de *S. cerevisiae* e *S. bayanus* foi possível seleccionar 18 e 4 sondas, respectivamente, a partir da base de dados local. No entanto, nenhuma destas foi validada por existirem outras espécies com as mesmas sequências na "nr". Um panorama semelhante foi observado para as regiões ITS2 das quatro espécies de leveduras. Foram seleccionadas sondas por pesquisa na base de dados local, mas as suas sequências não se revelaram específicas após o BLAST na base de dados universal.

Finalmente, foi possível encontrar sondas (quatro), direccionadas ao ITS1 capazes de identificarem inequivocamente *S. mikatae* em amostras biológicas laboratoriais, distinguindo-as de todas as outras espécies de fungos com sequência de DNA conhecida. O mesmo não foi conseguido para os ITS1 de *S. bayanus*, *S. cerevisiae* e *S. paradoxus*. A primeira por não ser única em nenhuma secção (de 15 nucleótidos) de entre as sequências da base de dados local e as duas últimas por não o serem em relação à base de dados universal do NCBI.

A Tabela 5 apresenta as sondas ou combinações de sondas obtidas após os processos de selecção explicados anteriormente. São igualmente apresentados as suas coordenadas de localização na sequência total do gene em questão.

Tabela 5 – Sequências, coordenadas do 1º nucl., % GC e Tm das sondas de 15 nucleótidos, seleccionadas nos genes de rRNA, para o desenho de um *chip* de diagnóstico para 10 espécies de fungos. Os genes identificados por combinação específica de 4 sondas estão assinalados com "*". Nestes casos, a identificação da espécie em questão apenas é validada quando se detecta hibridação com todas as 4 sondas em simultâneo. As coordenadas referentes aos alinhamentos dos genes 18S ou 28S (em Anexo), nas 10 espécies, estão assinaladas com "#".

Espécie	Gene	Coordenadas	Sequência (5'-3')	% G+C	Tm (ºC)
A. fumigatus	ITS1	83	CGCCGGGGAGGCCTT	80	54
	ITS2 *	101	TGGGGCTTTGTCACC	60	48
		106	CTTTGTCACCTGCTC	53	46
		113	ACCTGCTCTGTAGGC	60	48
		140	AGCCGACACCCAACT	60	48
	28S *	432	ACCAGACTCGCCCGC	73	52
		460	CATTCGTGCCGGTGT	60	48
		552	TTATAGCCGAGGGTG	53	46
		554	ATAGCCGAGGGTGCA	60	48
C. albicans	18S	695	CTTCTGGGTAGCCAT	53	46
	ITS1	72	CGCCAGAGGTCTAAA	53	46
	ITS2	122	GGTAACGTCCACCAC	60	48
	28S	451	CATGCTGCTCTCTCG	60	48
C. glabrata	ITS1	168	ACAAAGACCTGGGAG	53	46
	ITS2	170	GTTGATCTAGGGAGG	53	46
	28S	574	GAATACGGCCAGTCG	60	48
C. tropicalis	18S	657	CATCTTTCTGATGCG	47	44
	ITS1	57	GGGAGCAATCCTACC	60	48
	ITS2	110	TTTGCTAGTGGCCAC	53	46
	28S	513	GGAGAATTGCGTTGG	53	46
C. neoformans	18S *	556	CTCGTAGTCGAACTT	47	44
		566	AACTTCAGGTCTGGC	53	46
		603	GCACTGTCTTGCTGG	60	48
		618	ACCTTACCTCCTGGT	53	46
	ITS1	10	ATTGGACTTCGGTCC	53	46
	28S *	507	GTTCTGATCGGTGGA	53	46
		516	GGTGGATAAGGGCTG	60	48
		537	TGTGGCACTCTTCGG	60	48
		619	GGGTTCGCCCACGTT	67	50
S. mikatae	ITS1	85	AGTCCAGTGGGGCCT	67	50
S. pombe	18S	671	GCGTGTTTACTGGTC	53	46
	ITS1	211	TACGAGTGGATGATT	40	42
	ITS2	44	AGGTGTTGAACGAAA	40	42
	28S	473	TTCGCGAGACTATGC	53	46
Controlos Positivos	18S	779 #	GTGTTCAAAGCAGGC	53	46
	18S	1812 #	CAAGGTTTCCGTAGG	53	46
	28S	35 #	CGCTGAACTTAAGCA	47	44
	28S	687 #	CCCGTCTTGAAACAC	53	46
Controlos Negativos	–	–	GCATACTCGTCGCAG	60	48
	–	–	CGAGCAACCCGAGAT	60	48
	–	–	ACAGAGCTCCGGTAC	60	48
	–	–	TGGCCAATCCGCATC	60	48

As sondas apresentadas foram seleccionadas entre todas as de especificidade comprovada, com o critério de a sua %G+C se aproximar o mais possível do intervalo 53-

60%. As sondas que mais se distanciam deste valor são as de ITS1 e ITS2 de *S. pombe*, com 40% e ITS1 de *A. fumigatus*, com 80%. Neste caso, a sequência apresentada é mesmo a única sonda específica encontrada. Para os ITS's de *S. pombe*, todas as sondas específicas possuíam %G+C igual ou inferior a 40.

As sondas para os controlos negativos foram retiradas do genoma da planta *A. thaliana*. Estas sequências foram pesquisadas nas bases de dados "nr", subsecção *Fungi*, do NCBI e averiguou-se que não se encontram presentes em nenhum genoma dos fungos em estudo. Por esse motivo, em princípio, não ocorrerá hibridação no *chip* com sequências destas espécies, não se verificando sinal de fluorescência nos pontos de hibridação correspondentes.

Por outro lado, as sondas do controlo positivo, foram desenhadas de modo a serem complementares com sequências do rDNA de todas as 10 espécies, no propósito de hibridarem e originarem sinal fluorescente sempre que o *chip* seja incubado com amostras das espécies em estudo. Cada uma destas quatro sondas controlo corresponde a uma das quatro zonas de selecção de sondas específicas – ITS1, ITS2 e parte de 18S e 28S. A segunda sequência, com a coordenada 1812 do alinhamento de 18S, embora não pertença a ITS1, tem o seu alvo incluído na região amplificada (ver secção seguinte) para este *spacer*. O mesmo se aplica à terceira sequência (coordenada 35), dado que está direccionada para uma região do gene 28S que será amplificada conjuntamente com o ITS2.

Todas as sondas foram testadas, com o programa primers3.pl, para a determinação da possibilidade de formarem estrutura secundária por *self-anneal*. Para essa finalidade, este programa determinou as secções inversamente complementares dentro da cadeia de cada sonda. Para qualquer das sondas, não foram encontradas estas secções com extensões superiores a 4 bases, o que significa uma baixa probabilidade de ocorrência de estrutura secundária dentro da cadeia.

4. Amplificação da amostra por PCR.

Após ser definido o conjunto de sondas identificadoras a utilizar no *Funchip*, foi necessário ter em atenção a amostra que se vai utilizar para detectar a infecção. Ao utilizar sangue de um paciente para diagnosticar uma infecção por um fungo patogénico, é necessário ter em conta que a concentração de DNA do organismo infeccioso é muito

baixa em termos absolutos e relativamente à concentração de DNA humano. Deste modo, é necessário amplificar por PCR o material genético do fungo sem amplificar o do paciente.

Para maior simplicidade, tentou encontrar-se pares de *primers* que amplificassem simultaneamente as sequências dos genes de rRNA das espécies em estudo. Este estudo permitiu identificar 4 pares de *primers* que amplificam as sequências pertencentes a 18S, ITS1, ITS2 e 28S dos fungos *Aspergillus fumigatus*, *Candida albicans*, *Candida glabrata*, *Candida tropicalis*, *Cryptococcus neoformans*, *Saccharomyces bayanus*, *Saccharomyces cerevisiae*, *Saccharomyces mikatae*, *Saccharomyces paradoxus* e *Schizosaccharomyces pombe*.

Para desenhar os pares de *primers* essenciais à reacção de PCR, foi necessário, em primeiro lugar, saber quais os intervalos das sequências a amplificar. Estes correspondem às regiões do rDNA onde se localizam (os alvos para) as sondas seleccionadas anteriormente. De seguida, encontraram-se zonas adjacentes homólogas entre todas as 10 espécies de fungos e diferentes das do homem – recorrendo a alinhamentos múltiplos. Os *primers* de PCR foram seleccionados dentro dessas zonas recorrendo ao *script* primers3.pl.

Em relação a 18S e 28S (tamanho na ordem dos milhares de nucleótidos), apenas são amplificados intervalos mais pequenos (cerca de 200-300 nucl.). Estes intervalos foram determinados por um dos *scripts* que seleccionou as sondas, o compmultalign4.pl. As sequências de ITS1 e ITS2, são integralmente amplificadas, pois são relativamente pequenas.

Como foi referido anteriormente, as sondas desenhadas com base no gene 18S foram seleccionadas entre as posições 638 e 828 do alinhamento (Anexo 2). Pela observação das regiões adjacentes a esta secção foi possível encontrar as zonas conservadas 602-630 e 847-866. Nestas duas regiões, a sequência das 10 espécies fúngicas é igual enquanto que a do *H. sapiens* apresenta um e quatro nucleótidos diferentes, respectivamente (ver alinhamento da Figura 41). Recorrendo ao *script* primers3.pl, caracterizaram-se todos os possíveis *primers* de 15 nucleótidos pertencentes a estas zonas.

```
albicans        GTATATTAAAGTTGTTGCAGTTAAAAAGC...GGTTCTATTTTGTTGGTTTCTAGGA
tropicalis      GTATATTAAAGTTGTTGCAGTTAAAAAGC...GGTTCTATTTTGTTGGTTTCTAGGA
bayanus         GTATATTAAAGTTGTTGCAGTTAAAAAGC...GGTTCTATTTTGTTGGTTTCTAGGA
paradoxus       GTATATTAAAGTTGTTGCAGTTAAAAAGC...GGTTCTATTTTGTTGGTTTCTAGGA
mikatae         GTATATTAAAGTTGTTGCAGTTAAAAAGC...GGTTCTATTTTGTTGGTTTCTAGGA
cerevisiae      GTATATTAAAGTTGTTGCAGTTAAAAAGC...GGTTCTATTTTGTTGGTTTCTAGGA
glabrata        GTATATTAAAGTTGTTGCAGTTAAAAAGC...GGTTCTATTTTGTTGGTTTCTAGGA
pombe           GTATATTAAAGTTGTTGCAGTTAAAAAGC...GGTTCTATTTTGTTGGTTTCTAGGA
fumigatus       GTATATTAAAGTTGTTGCAGTTAAAAAGC...GGTTCTATTTTGTTGGTTTCTAGGA
neoformans      GTATATTAAAGTTGTTGCAGTTAAAAAGC...GGTTCTATTTTGTTGGTTTCTAGGA
sapiens         GTATATTAAAGTTGCTGCAGTTAAAAAGC...GGTTCTATTTTGTTGGTTTTCGGAA
                ************** **************   ******************   * *
```

Figura 41 – Alinhamento das duas regiões, do gene 18S, adjacentes à zona onde foram seleccionadas as sondas, nas 10 espécies de fungos e em *H. sapiens*. O *primer* 5' e o *primer* 3' foram seleccionados na primeira e na segunda regiões, respectivamente (separadas por "...").

Para seleccionar os *primers*, para amplificar a região do gene 28S com os alvos para as sondas desenhadas, seguiu-se o mesmo princípio que para o gene anterior. As zonas adjacentes consideradas foram 329-344 e 687-716 do alinhamento (Anexo 4). Embora a sequência da segunda zona seja igual no genoma dos fungos e do homem, a sequência da primeira zona apresenta grande variabilidade entre *H. sapiens* e as 10 espécies fúngicas, o que não permite a completa amplificação do DNA daquele.

A amplificação da totalidade do ITS1 é possível através de um *primer* 5' seleccionado na região conservada 1839-1853 no término do gene 18S e de um *primer* 3' determinado na região conservada 32-46 no início do gene 5.8S (Alinhamento no Anexo 3). Para a escolha dos *primers* necessários à amplificação do ITS2 foram tidas em conta as regiões conservadas 47–63 do 5.8S e 35–61 da parte inicial do 28S.

Os *primers*, de 15 nucleótidos, seleccionados para a amplificação por PCR de 18S, ITS1, ITS2 e 28S estão apresentados na Tabela 6, bem como a sua temperatura de fusão. Estão igualmente indicadas as extensões (em bp) dos quatro produtos de PCR previstos para cada espécie.

Tabela 6 – *Primers* seleccionados para a amplificação por PCR de ITS1, ITS2 e secções de 18S e 28S nas 10 espécies em simultâneo. Para cada espécie/gene é indicada a extensão (bp) do respectivo produto de PCR. Na sequência dos *primers*, os nucleótidos a negrito e sublinhado representam as posições não conservadas entre as espécies de fungos em estudo e *H. sapiens*.

	18S	ITS1	ITS2	28S
primer 5'	TTAAAGTTG**T**TGCAG (40 ºC)	G**GTC**AT**T**AGAGGAA (42 ºC)	CAGCG**AAA**TGCGA**T**A (44 ºC)	C**AT**C**TA**AAG**C**T**AA**AT (38 ºC)
primer 3'	T**C**C**TAG**AAACCAACA (42 ºC)	CGTTCTTCATCGA**TG** (44 ºC)	**TG**C**TTAAGT**TC**AG**CG (44 ºC)	TCCGTGTTTCAAGAC (44 ºC)
A. fumigatus	246	295	321	340
C. albicans	233	249	305	342
C. glabrata	246	513	384	351
C. tropicalis	231	249	295	340
C. neoformans	247	232	342	366
S. bayanus	246	469	385	342
S. cerevisiae	246	472	387	342
S. mikatae	246	471	385	342
S. paradoxus	246	473	386	342
S. pombe	258	531	454	359

Os *primers* estão indicados na orientação 5'-3'.

A selecção de *primers*, foi realizada obedecendo a vários critérios. Por exemplo, após verificar que as suas sequências pertenciam a regiões adjacentes às de selecção das sondas, que eram conservadas nas 10 espécies de fungos e que eram parcialmente diferentes das de *H. sapiens*, foi necessário encontrar sequências de 15 nucleótidos com aproximadamente a mesma Tm. O *primer* 5' do gene 28S foi o que mais se distanciou da média das temperaturas de fusão de todos os *primers*, com um valor de 38 ºC.

O programa primers3.pl determinou as extensões máximas das secções inversamente complementares dentro de todas as cadeias dos *primers*, não se inferindo qualquer possibilidade de formarem estrutura secundária. A maior probabilidade de auto-hibridação ocorre no *primer* 3' do ITS2, entre as secções GCT (posições 2-4) e AGC (12-14). Por último, não se identificou um grau de complementaridade significativo entre os vários *primers* seleccionados. Os máximos observados entre dois *primers* é de apenas 6 nucleótidos complementares (por exemplo, entre ATTTAG do *primer* 5' de ITS1 e CTAAAT do *primer* 5' de 28S).

A sequência do *primer* 3' de 28S, como foi referido, não apresenta nenhum nucleótido diferente em relação à sequência homóloga de *H. sapiens*. Este facto poderá afectar o rendimento da amplificação das sequências de DNA deste gene nos fungos, por competição com o DNA do homem. Uma alternativa seria utilizar o *primer* 5'-CCCT**ATT**CAGGCATA-3' (881-895 do alinhamento - Anexo 4), que apresenta 3

nucleótidos de diferença (a negrito e sublinhado). No entanto, com este *primer* os respectivos produtos de PCR sofreriam um incremento de aproximadamente 180 bp.

Observa-se alguma diversidade a nível da extensão dos produtos de PCR, com um mínimo de 231 bp no gene 18S de *C. tropicalis* e um máximo de 531 bp no ITS1 de *S. pombe*. Estes valores introduzem algum *bias* no rendimento das várias reacções de amplificação e possivelmente uma diferenciação dos constrangimentos espaciais aquando da hibridação com as sondas do chip. Porém, não é esperado que ocorram alterações significativas dos resultados finais, dado que a ordem de grandeza de todos os produtos de PCR é semelhante.

Capítulo IV: Discussão

A aplicação da tecnologia de *chips* de DNA no diagnóstico molecular de doenças infecciosas permite o desenvolvimento de metodologias simples e rápidas de identificação das espécies patogénicas responsáveis. Metodologias deste tipo tem vindo a ser extensivamente implementadas no diagnóstico de infecções provocadas por espécies de bactérias e vírus. Neste projecto, desenvolveram-se estratégias de desenho de *chips* de DNA de diagnóstico para a identificação de fungos patogénicos presentes em amostras clínicas.

1. Estratégias de desenho de sondas para *chips* de DNA de diagnóstico molecular.

Na construção de um *chip* de diagnóstico, o principal factor a considerar é a constituição das sondas a depositar em cada ponto de hibridação. Neste trabalho, foram desenvolvidas duas estratégias no desenho de sondas oligonucleotídicas. A primeira, baseou-se na elevada capacidade de pesquisa do algoritmo BLAST, para seleccionar sondas com um máximo de individualidade em relação a sequências de outras espécies. A segunda estratégia, delineada para sondas mais pequenas, qualifica a capacidade específica de uma sequência pelos nucleótidos das posições centrais. Ambas as aproximações, assumem que cada sonda fixada no suporte sólido do *chip* de DNA apenas poderá hibridar com a sequência alvo para a qual foi desenhada. Sequência essa que existe unicamente numa das espécies de fungos que se pretendem identificar. A todas as outras sequências existentes na amostra, mesmo as parcialmente complementares, não será possível a formação de uma estrutura de dupla cadeia estável com a sonda.

A primeira estratégia foi aplicada no desenho de sondas de 50 nucleótidos cada. A segunda estratégia foi usada para desenhar sondas com um secção de 15 nucleótidos específicos. A extensão dos oligonucleótidos depositados na superfície do *chip* condiciona quer a especificidade quer a sua sensibilidade, ou seja, a capacidade de hibridar com pequenas concentrações de ácidos nucleicos da amostra. Sondas de oligonucleótidos maiores (50-70 nucleótidos) permitem detectar mais facilmente a sequência alvo de entre todos os ácidos nucleicos de uma amostra complexa do que sondas de oligonucleótidos mais pequenos [99, 127]. No entanto, estas são mais específicas, pois são capazes de distinguir duas sequências com um único nucleótido de diferença. Pelo contrário, sondas

de 50 nucleótidos podem hibridar com sequências que apresentem um grau de semelhança superior a 75% com a sua sequência alvo, resultando em hibridação cruzada [127]. Devido a estas propriedades, diferentes estudos utilizam sondas de diferentes extensões.

A primeira estratégia de desenho de *chips* de DNA de diagnóstico para identificação de fungos patogénicos foi aplicada na determinação de dois conjuntos de sondas, cada uma com 50 nucleótidos, capazes de distinguirem as leveduras *C. albicans* e *S. cerevisiae*. Esta estratégia baseia-se no mínimo de complementaridade que pode existir entre cada sonda e segmentos de sequência que são encontrados noutras espécies. A "qualidade" desta possível hibridação não específica é medida em termos de temperatura de fusão ($Tm_{máx\_inesp}$). Quanto menor for a $Tm_{máx\_inesp}$ calculada para uma sequência, menos previsível é a ocorrência dessa hibridação inespecífica, logo maiores são as probabilidades de funcionar como sonda. Se, por outro lado, uma candidata a sonda identificativa de uma determinada espécie tiver um valor de $Tm_{máx\_inesp}$ igual à sua Tm, isso indica que existe outra espécie com uma sequência totalmente igual (e outra totalmente complementar) à sua. Neste caso extremo, se a sonda fosse colocada num *chip*, iria hibridar com as sequências das duas espécies com a mesma magnitude, o que obviamente não é o pretendido.

As sondas seleccionadas para *C. albicans* e *S. cerevisiae* apresentam valores de $Tm_{máx\_inesp}$ entre os 20 e 28 ºC. Em comparação com a Tm de cada sonda, na gama 118-156 ºC, estes valores permitem deduzir que não ocorrerá hibridação cruzada no *chip* de DNA, desde que as condições de incubação não sejam demasiado permissivas. Esta estratégia permite desenhar sondas em regiões do genoma que apresentem grande variabilidade interespecífica, como é o caso do ITS1 e do ITS2. Caso se utilize um gene com sequência muito conservada, possuindo baixa frequência de nucleótidos variáveis, será difícil obter sondas específicas com menos de 75% de semelhança entre espécies.

Entre duas sequências parcialmente complementares, especialmente se forem pouco extensas, a posição dos *mismatches* é essencial na ocorrência ou não de hibridação. A existência de nucleótidos não complementares nas posições centrais é mais desestabilizadora do que nos casos em que eles se encontram nas extremidades [95]. Este fenómeno explica-se por ser menos favorável termodinamicamente a formação de uma estrutura estável com tensões disruptivas no meio da dupla hélice [128-130]. Este princípio

é a base da segunda estratégia de design de sondas apresentada nesta tese. A condição *sine qua non* respeitada é a existência de nucleótidos únicos nas posições 7 e/ou 8 e/ou 9 de cada sonda de 15 nucleótidos, em relação a sequências de outras espécies. Deste modo será evitada a ocorrência de sinal resultante de falsos positivos nos pontos de hibridação do *chip*. Este critério permitiu a determinação de sondas para sete das dez espécies em estudo.

Como seria de esperar, das espécies em estudo, as mais distantes filogeneticamente, nomeadamente *S. pombe*, *A fumigatus* e *C. neoformans* (ver alinhamentos em Anexo), são aquelas para as quais foi mais fácil encontrar sondas específicas a nível local (considerando apenas as 10 espécies em estudo). No entanto, a pesquisa na base de dados universal "nr" instalada nos servidores do NCBI conduziu à rejeição de grande parte dessas sondas nas espécies *A. fumigatus* e *C. neoformans*, dado que eram sequências partilhadas por outras espécies não integradas neste estudo. Ainda assim, cada uma destas espécies será representada no *chip* por uma sonda específica para ITS1 e duas combinações de quatro sondas. As três espécies do género *Candida* revelaram-se suficientemente distantes evolutivamente, entre elas e entre todas as outras, pois foram desenhadas sondas específicas para todos os genes, com a excepção do 28S de *C. glabrata*.

A análise das espécies do género *Saccharomyces* revelou-se, em larga medida, infrutífera, como demonstram os resultados obtidos. Neste caso, as causas da impossibilidade de descobrir sondas específicas têm origem quer na similaridade das quatro espécies analisadas localmente, isto é: na elevada identidade entre as sequências dos seus genes, quer na existência de outras *Saccharomyces* com sequências de rDNA semelhantes. No entanto, a obtenção da sonda para *S. mikatae* na região ITS1 é bastante encorajadora sobre as possibilidades de aplicação da tecnologia de *microarrays* e desta estratégia de desenho de sondas, na identificação diferenciada de espécies extremamente homólogas.

Convém ainda sublinhar que nem todas as espécies estudadas são capazes de iniciar um processo infeccioso em hospedeiros humanos. Das 10 espécies, todas aquelas com propriedades patogénicas reconhecidas estão representadas no *chip* com sondas representativas de, no mínimo, 75% dos genes estudados. É o caso de *A. fumigatus*, *C. albicans*, *C. glabrata*, *C. tropicalis* e *C. neoformans*. Em relação às quatro Saccharomyces, a sua utilização neste estudo tinha como objectivo a validação desta estratégia de desenho de *chips* de DNA em casos extremos, o que foi parcialmente conseguido.

As observações anteriores podem ser extrapoladas para um grande número de espécies de fungos patogénicos, dado que a estratégia de desenho aqui desenvolvida poderá ser utilizada para qualquer genoma. É muito pouco provável que duas espécies patogénicas possuam um grau de semelhança entre as suas sequências de DNA equivalente à que existe, por exemplo, entre *S. cerevisiae* e *S. bayanus*. Por outro lado, mesmo que isso se verifique, as características de virulência, os processos infecciosos e a sensibilidade a antifúngicos desses agentes tenderão a ser equivalentes. Nessas situações, a tecnologia de *chips* de DNA, caso não consiga a identificação diferenciada das duas espécies, obterá, no mínimo, um rápido diagnóstico parcial, que permitirá iniciar imediatamente o tratamento clínico adequado. Considerando a baixa fiabilidade das alternativas tradicionais na identificação de espécies tão semelhantes e o tempo necessário à realização do diagnóstico, os *chips* de DNA representam um avanço significativo no diagnóstico de microrganismos patogénicos.

Foram realizados estudos semelhantes com a utilização de sondas pequenas capazes de discriminar diferenças de poucos nucleótidos entre sequências diferentes. Oligonucleótidos específicos de 17-35 foram utilizados como sondas num chip de diagnóstico capaz de identificar quatro espécies de bactérias patogénicas do género *Campylobacter*. Para cada espécie, foram desenhadas várias sondas específicas em cada um dos seis genes marcadores estudados (fur, glyA, ceuB-C, cdts e fliY), o que permitiu a melhor distinção entre as várias espécies [91]. Um *Chip* de DNA constituído por sondas de 18 nucleótidos foi utilizado com sucesso na identificação de procariótas redutores de sulfato em amostras ambientais e clínicas. Estas sondas possuíam os seus alvos no gene de rRNA 16S das várias espécies e foram desenhadas de modo a terem o maior número possível de *mismatches* nas posições centrais em relação a todas as espécies excepto a alvo [96]. Bodrossy e colaboradores (2003) utilizaram sondas de oligonucleótidos pequenos (15 a 26 nucl de extensão) direccionadas ao gene pmoA para identificarem eficazmente várias espécies de bactérias metanotróficas. Com base nos resultados obtidos, concluíram que a existência de *mismatches* nas extremidades não altera significativamente a hibridação ao contrário dos existentes no meio [95]. Outro estudo, utilizou um *chip* comercial fornecido pela companhia *Affymetrix Corporation* com sondas de 20 nucleótidos para identificar bactérias em culturas puras e numa amostra de ar. Com estas sondas pequenas,

direccionadas para o gene 16S do rRNA, foi possível realizar a identificação correcta de um grande número de espécies presentes nas amostras [94].

As sondas do controlo negativo seleccionadas para este *chip* de diagnóstico foram extraídas do gene *ribulose bisphosphate carboxylase/oxygenase* (RuBisCO) *activase* da planta *A. thaliana*. Este gene está envolvido nos processos de fotossíntese e como tal não se encontra presente no genoma dos fungos patogénicos em estudo. Portanto, é esperado que a sequência de 15 nucleótidos de cada sonda não hibride com as sequências das amostras clínicas a utilizar. A observação de qualquer fluorescência nos pontos de hibridação correspondentes, resultante de hibridação parcial e/ou deficiências na lavagem, será utilizada para efeitos de normalização do sinal. Qualquer dos pontos de hibridação com sondas específicas que apresente um sinal semelhante não será, evidentemente, reconhecido como "positivo".

Os controlos positivos foram, tal como os *primers*, seleccionados a partir de zonas conservadas em todas as espécies. Todas as sondas são também complementares com o DNA humano, com excepção da sonda com a coordenada inicial na posição 35 do alinhamento do 28S. Essa mesma sequência é complementar da sequência do *primer* 3' de ITS2, dado que não havia outra região conservada dentro do produto de PCR. Nestas condições, os pontos de hibridação com estas quatro sondas apresentarão sinal fluorescente após a incubação com amostra clínica amplificada por PCR. Este sinal, tal como o dos controlos negativos é importante para a determinar a qualidade dos sinais detectados nos outros pontos de hibridação.

Para a obtenção de uma eficiência de hibridação semelhante em todas as sondas e consequentemente de um sinal homogéneo entre todos os pontos de hibridação com fluorescência, foi necessário controlar os parâmetros conteúdo G+C, temperatura de fusão e auto-complementaridade das sondas definidas como específicas. Para determinados genes de algumas espécies, especialmente do segundo *chip* de DNA, não foi possível maximizar esta optimização, ao nível do conteúdo G+C, devido ao pequeno número de sondas disponíveis após os testes de especificidade. Por este motivo, é aconselhável realizar a incubação entre amostra e *chip* em condições pouco restritivas.

A eficiência de uma sonda pode também ser influenciada por factores espaciais. O suporte sólido do *chip* de DNA pode interferir nas reacções de hibridação. Por outro lado, a concentração elevada dos oligonucleótidos imobilizados pode dificultar a sua acessibilidade por parte das sequências da amostra. Para minimizar estes obstáculos, é possível aplicar caudas às sondas de um *chip* de DNA, para se localizarem entre o oligonucleótido específico e a superfície do *chip*. Estes modificações têm objectivos unicamente espaciais pois não intervêm directamente na hibridação, ou seja, não são desenhadas para serem complementares a outras sequências. Normalmente são constituídas por sequências repetidas de um único nucleótido. Outros estudos utilizaram com sucesso caudas constituídos por sequências de poli-A de tamanhos variáveis (6 a 18-mer) [131] e sequências de 15 poli-T [96]. No *chip* de DNA desenhado na segunda parte desta tese está prevista a utilização, em cada sonda, de uma cauda de 15 timinas para fazer a ponte entre a superfície sólida e o oligonucleótido específico.

2. Amplificação das sequências da amostra por PCR.

Para aumentar a concentração do DNA fúngico existente numa amostra biológica particular, foi delineada uma estratégia de amplificação por PCR *Multiplex*. A principal condicionante nesta estratégia é a existência de regiões conservadas entre as várias espécies que se encontrem adjacentes em ambos os flancos da região utilizada para seleccionar as sondas. Neste estudo, foi possível determinar unicamente quatro pares de *primers* que permitem a amplificação das secções pretendidas do rDNA nas 10 espécies simultaneamente. Esta possibilidade simplifica a execução laboratorial da reacção de PCR, pois a multiplicação de pares de *primers* na mesma câmara de PCR aumenta a complexidade da reacção e dos produtos formados e a possibilidade de hibridação entre *primers* parcialmente complementares. A determinação de *primers* com sequências exactamente complementares às de todas as espécies em estudo, torna desnecessário a utilização de *primers* degenerados, que poderiam originar diferentes quantidades do produto de PCR formado para cada espécie.

Posteriormente, outro factor considerado foi a existência de nucleótidos não conservados em relação a *H. sapiens*, na sequência dos *primers* escolhidos. Apenas se considerou esta espécie porque numa amostra clínica a concentração de DNA humano será obviamente predominante e sempre muito superior à de qualquer organismo infeccioso.

Este critério não foi obedecido apenas na primeira opção do *primer* 3' de 28S. No entanto, espera-se que a amplificação da sequência deste gene no genoma humano não se conclua completamente, dado que o *primer* 5' apresenta quase 50% de nucleótidos não conservados. De qualquer forma, se laboratorialmente se concluir que estas propriedades estão a afectar os resultados pretendidos, poderá recorrer-se ao *primer* 3' alternativo, tendo em atenção que o produto de PCR será maior. Em relação a outras espécies igualmente presentes, poderá ocorrer amplificação das suas sequências, mas não é esperado que prejudiquem em larga escala a eficiência da amplificação do rDNA dos fungos em estudo.

A amplificação por PCR de sequências de DNA de determinadas espécies em amostras biológicas, antes de incubação com um *microarray*, foi já aplicada em vários estudos, amplificando-se simultaneamente os genes 16S [89, 93, 94, 96], 23S [92] ou outros genes marcadores [87, 88, 90, 91, 95, 98, 99] de várias espécies de bactérias. Também em experiências com vírus foram amplificadas por RT-PCR sequências alvo para as sondas dos *chips* [83-86]. A combinação das duas técnicas é muito eficaz neste tipo de estudos de diagnóstico, permitindo a detecção e discriminação entre organismos patogéneos numa amostra biológica complexa.

A utilização da técnica de PCR neste trabalho serve essencialmente para amplificação de DNA. No entanto, a sua aplicação leva, paralelamente, à discriminação de todas as espécies que não possuam sequências complementares à dos *primers*, o que pode ser considerado com um primeiro passo na identificação das espécies. Em certos estudos, porém, os protocolos de PCR são projectados para realizarem por si só a identificação das espécies presentes numa amostra. Isto é possível recorrendo a pares de *primers* direccionados a determinadas espécies, tal como nas sondas específicas. Se numa reacção de PCR for introduzido um par de *primers* específico, apenas irá ocorrer amplificação de DNA se a espécie correspondente estiver presente na amostra. A detecção e extensão do produto de PCR formado é posteriormente realizada através de uma electroforese em gel. A identificação de mais do que uma espécie na mesma reacção de PCR requer a introdução de outros tantos pares de *primers*, o que torna o protocolo e a análise mais complexos. Contudo, já foi possível identificar 14 espécies de leveduras, incluindo *C. albicans*, *C. glabrata*, *C. tropicalis* e *Pichia guilliermondii*, através deste método [132]. Em cada mistura de reacção de PCR foram incluídos um par de *primers* específicos e um par universal para controlo positivo. Desta forma, em cada reacção foi determinado se uma das

espécies se encontra presente na amostra através da identificação da banda correspondente ao produto de PCR esperado. A desvantagem deste método de diagnóstico é a dificuldade de identificar mais do que uma espécie numa única reacção e ainda na análise automática dos resultados no gel.

Alternativamente, pode ser realizada a incubação de *chips* de DNA com amostra não amplificada por PCR [97, 133, 134], utilizando directamente moléculas de rRNA que existem em relativa abundância nas células. Contudo, este tipo de procedimentos tem muito ruído devido à densidade de células e de rRNA, entre espécies contaminantes, ser diferente. Isto é, espécies que se encontrem em maior grau de proliferação dão um sinal forte, enquanto espécies raras produzem um sinal ténue ou mesmo imperceptível.

3. Estrutura secundária.

A formação de estrutura secundária é um factor a considerar no desenho de sondas de oligonucleótidos. Por esse motivo, neste estudo foram seleccionadas sondas com nível ínfimo de auto-complementaridade para evitar esse fenómeno indesejado. A formação de ganchos, haste-laços ou outras estruturas semelhantes é igualmente possível nas cadeias da sequência alvo, especialmente se se tratar de RNA. Para estes casos, uma das estratégias desenvolvidas foi a utilização de um segundo tipo de sondas, sondas detectoras, que são misturadas na solução da amostra biológica [134]. A sequência de cada sonda detectora é desenhada, tal como as das sondas normais que são fixadas no *chip*, de modo a ser complementar com um segmento da cadeia alvo de RNA da amostra. No entanto, a sonda normal e a sonda detectora hibridam com segmentos diferentes, não sobreponíveis, mas em posições relativamente próximas. Deste modo, para cada cadeia alvo de RNA da amostra é desenhada uma sonda normal e uma sonda detectora que preenchem os locais de ligação da cadeia alvo nessa região alargada. Portanto, a adição de uma sonda detectora tem por objectivo diminuir as probabilidades de formação, por auto-complementaridade, de estrutura secundária estável nessa região do RNA, o que poderia impedir a acessibilidade entre as bases complementares da sonda normal e do alvo. Este problema não foi considerado neste trabalho, dado que está previsto que a amostra a incubar com o *chip* seja constituída por cadeias de DNA, um ácido nucleico muito menos disponível para este tipo de formação. Não obstante, é importante sublinhar que a amostra terá que ser submetida a

condições de desnaturação das duplas cadeias existentes, seja pelo aumento da força iónica, da concentração de formamida ou da temperatura da solução.

Capítulo V: Conclusões e Trabalho Futuro

As infecções provocadas por fungos patogénicos são, cada vez mais, uma preocupação da comunidade médica, devido à sua alta taxa de incidência sobretudo entre indivíduos imunodeprimidos, como é o caso dos doentes com SIDA. Estas patologias podem, em determinadas condições, ter consequências fatais, mas podem ser evitadas através de um tratamento clínico atempado. Por conseguinte, urge desenvolver novas metodologias de diagnóstico capazes de identificar num curto espaço de tempo a espécie responsável pela infecção, contornando a morosidade das técnicas tradicionais.

Os *chips* de DNA são uma tecnologia poderosa de identificação de ácidos nucleicos em amostras biológicas. A sua aplicação pode ser estendida ao diagnóstico de agentes infecciosos em tecidos retirados de pacientes hospitalares. Os princípios desta tecnologia foram aplicados com sucesso no diagnóstico de vírus e bactérias, podendo igualmente ser utilizados para o caso de fungos patogénicos

Com um único *chip* de diagnóstico, é possível identificar várias espécies simultaneamente. Esta possibilidade traduz-se numa fácil utilização e numa baixa relação qualidade/preço.

Neste trabalho foram desenvolvidas duas estratégias de desenho de sondas, com diferentes tamanhos, para *chips* de DNA de diagnóstico. Os protocolos descritos baseiam-se na recolha de sequências de DNA disponíveis em bases de dados online, em algoritmos capazes de comparar essas sequências e determinar as suas singularidades e em programação informática. A selecção das sondas faz-se de modo semi-automático pela execução dos programas implementados.

No planeamento e desenho de um *chip* de diagnóstico, é fundamental considerar a filogenia molecular das espécies que se pretendem analisar. Espécies muito próximas evolutivamente possuem estruturas moleculares homólogas, dificultando a sua identificação diferenciada através das sondas de um *chip*. Neste trabalho foi possível distinguir a espécie *S. mikatae* de outras espécies de *Saccharomyces* com extrema proximidade filogenética entre si. No entanto, não foi possível distinguir outras 3 espécies do mesmo género.

Os genes do rRNA são muito úteis neste tipo de estudo, dado que apresentam significativa variabilidade entre espécies. Contudo, em trabalhos futuros, deverão ser

adicionalmente analisados outros genes, ou mesmo zonas não codificantes, no propósito de se determinarem regiões com uma taxa superior de nucleótidos não conservados. Essas regiões funcionarão como alvos para sondas específicas que poderão permitir a distinção entre espécies filogeneticamente muito próximas.

Neste trabalho, foi igualmente desenvolvida uma estratégia para amplificar o DNA fúngico das amostras clínicas através de PCR *Multiplex*. A utilização de um único par de *primers* permite a amplificação simultânea de cada gene, em todas as espécies em estudo.

Da necessidade de zonas conservadas para desenhar os primers e zonas não conservadas para desenhar as sondas advém um dos principais desafios deste trabalho. No rDNA foi possível obedecer a estes critérios com relativa facilidade. Para outras regiões do genoma em que tal não seja possível, será necessário recorrer a *primers* degenerados ou um maior número de pares de *primers* para a amplificação da amostra.

Os sistemas informáticos implementados poderão, no futuro, ser optimizados e integrados numa única aplicação homogénea e mais automatizada. Poderá ser adicionada uma funcionalidade de *download* e tratamento automático das sequências. A selecção dos *primers* deverá igualmente ser integrada com a selecção das sondas, de forma a que todas as combinações possíveis sejam avaliadas automaticamente. A interface gráfica com o utilizador poderá ser desenvolvida na continuação daquela que aqui foi desenvolvida em PERL, HTML e CGI ou recorrendo a outras linguagens de programação.

Globalmente, as estratégias desenvolvidas e os *chips* de DNA desenhados neste trabalho implicam um procedimento de diagnóstico de infecções de origem fúngica muito mais rápido e com menor margem de erro do que os métodos tradicionais de diagnóstico.

Anexos

Anexo 1

Na Tabela 7 estão indicados os códigos de acesso das entradas das bases de dados biológicas online referentes às sequências utilizadas na montagem das sequências completas dos genes de rRNA.

Tabela 7 – Códigos de acesso das sequências analisadas neste trabalho.

Espécie	Códigos de acesso
A. fumigatus	M55626, AF455431, AF109336, AJ438344
C. albicans	M60302, AF217609, AY196001, CAL551313, L07796
C. glabrata	X51831, AY198398,
C. tropicalis	M55527, AF321539, L47112, U45749, AF257268
C. neoformans	L05428, M55625, AF444326, L14067, L14068
S. bayanus	X97777, D89887, AY048156
S. cerevisiae	NC_001144*, Z75578, U53879
S. mikatae	AJ271812, AY130308, AF398479
S. paradoxus	X97806, D89890, AF005703, AY048155
S. pombe	Z19578, V01361, X58056

∗ - código de acesso do GenBank para a sequência completa do cromossoma XII.

Anexo 2

Alinhamento múltiplo de sequências do gene de rRNA 18S das espécies *Aspergillus fumigatus*, *Candida albicans*, *Candida glabrata*, *Candida tropicalis*, *Cryptococcus neoformans*, *Saccharomyces bayanus*, *Saccharomyces cerevisiae*, *Saccharomyces mikatae*, *Saccharomyces paradoxus* e *Schizosaccharomyces pombe* realizado pelo algoritmo Clustal através da interface gráfica ClustalX (versão 1.81).
O símbolo "*" por baixo de uma coluna de nucleótidos indica que essa posição é conservada nas 10 espécies.

```
C. albicans     TATCTGGTTGATCCTGCCAGTAGTCATATGCTTGTCTCAAAGATTAAGCCATGCATGTCT 60
C. tropicalis   TATCTGGTTGATCCTGCCAGTAGTCATATGCTTGTCTCAAAGATTAAGCCATGCATGTCT
S. bayanus      ---CTGGTTGATCCTGCCAGTAGTCATATGCTTGTCTCAAAGATTAAGCCATGCATGTCT
S. paradoxus    ---CTGGTTGATCCTGCCAGTAGTCATATGCTTGTCTCAAAGATTAAGCCATGCATGTCT
S. mikatae      ---CTGGTTGATCCTGCCAGTAGTCATATGCTTGTCTCAAAGATTAAGCCATGCATGTCT
S. cerevisiae   TATCTGGTTGATCCTGCCAGTAGTCATATGCTTGTCTCAAAGATTAAGCCATGCATGTCT
C. glabrata     TATCTGGTTGATCCTGCCAGTAGTCATATGCTTGTCTCAAAGATTAAGCCATGCATGTCT
S. pombe        TACCTGGTTGATCCTGCCAGTAGTCATATGCTTGTCTCAAAGATTAAGCCATGCATGTCT
A. fumigatus    AACCTGGTTGATCCTGCCAGTAGTCATATGCTTGTCTCAAAGATTAAGCCATGCATGTCT
C. neoformans   ------------------------------------------------------------


C. albicans     AAGTATAAGCAATTT-ATACAGTGAAACTGCGAATGGCTCATTAAATCAGTTATCGTTTA 120
C. tropicalis   AAGTATAAGCAATTT-ATACAGTGAAACTGCGAATGGCTCATTAAATCAGTTATCGTTTA
S. bayanus      AAGTATAAGCAATTT-ATACAGTGAAACTGCGAATGGCTCATTAAATCAGTTATCGTTTA
S. paradoxus    AAGTATAAGCAATTT-ATACAGTGAAACTGCGAATGGCTCATTAAATCAGTTATCGTTTA
S. mikatae      AAGTATAAGCAATTT-ATACAGTGAAACTGCGAATGGCTCATTAAATCAGTTATCGTTTA
S. cerevisiae   AAGTATAAGCAATTT-ATACAGTGAAACTGCGAATGGCTCATTAAATCAGTTATCGTTTA
C. glabrata     AAGTATAAGCAATTT-ATACAGTGAAACTGCGAATGGCTCATTAAATCAGTTATCGTTTA
S. pombe        AAGTATAAGCAATTTTGTACTGTGAAACTGCGAATGGCTCATTAAATCAGTTATCGTTTA
A. fumigatus    AAGTATAAGCAATTT-ATACGGTGAAACTGCGAATGGCTCATTAAATCAGTTATCGTTTA
C. neoformans   ----------GAATTCATACTGTGAAACTGCGAATGGCTCATTAAATCAGTTATAGTTTA
                          * **  *** ******************************** *****

C. albicans     TTTGATAGTACCTT-ACTACTTGG-ATAACCGTGGTAATTCTAGAGCTAATACATGCTTA 180
C. tropicalis   TTTGATAGTACCTT-ACTACTTGG-ATAACCGTGGTAATTCTAGAGCTAATACATGCTTA
S. bayanus      TTTGATAGTTCCTTTACTACATGGTATAACTGTGGTAATTCTAGAGCTAATACATGCTTA
S. paradoxus    TTTGATAGTTCCTTTACTACATGGTATAACTGTGGTAATTCTAGAGCTAATACATGCTTA
S. mikatae      TTTGATAGTTCCTTTACTACATGGTATAACTGTGGTAATTCTAGAGCTAATACATGCTTA
S. cerevisiae   TTTGATAGTTCCTTTACTACATGGTATAACTGTGGTAATTCTAGAGCTAATACATGCTTA
C. glabrata     TTTGATAGTTCCTTTACTACATGGTATAACTGTGGTAATTCTAGAGCTAATACATGCTTA
S. pombe        TTTGATAGTACCTCAACTACTTGG-ATAACCGTGGTAATTCTAGAGCTAATACATGCTAA
A. fumigatus    TTTGATAGTACCTT-ACTACATGG-ATACCTGTGGTAATTCTAGAGCTAATACATGCTAA
C. neoformans   TTTGATGGTATCTT-GCTACATGG-ATAACTGTGGTAATTCTAGAGCTAATACATGCTGA
                ****** **  **   **** *** *** * ***************************** *

C. albicans     AAA-TCCCGACTGTTT-GGAAGGGATGTATTTATTAGATAAAAAATCAATGCCTTCGGGC 240
C. tropicalis   AAA-TCCCGACTGTTT-GGAAGGGATGTATTTATTAGATAAAAAATCAATGTCTTCGGAC
S. bayanus      AAA-TCTCGACCCTTT-GGAAGAGATGTATTTATTAGATAAAAAATCAATGTCTTCGGAC
S. paradoxus    AAA-TCTCGACCCTTT-GGAAGAGATGTATTTATTAGATAAAAAATCAATGTCTTCGGAC
S. mikatae      AAA-TCTCGACCCTTT-GGAAGAGATGTATTTATTAGATAAAAAATCAATGTCTTCGGAC
S. cerevisiae   AAA-TCTCGACCCTTT-GGAAGAGATGTATTTATTAGATAAAAAATCAATGTCTTCGGAC
C. glabrata     AAA-TCTCGACCTCTT-GGAAGAGATGTATTTATTAGATAAAAAATCAATGTCTTCGGAC
S. pombe        AAA-TCCCGACTTTTTTGGAAGGGATGTATTTATTAGATAAAAAACCAATGCCTTCGGGC
A. fumigatus    AAA-CCTCGACTTC---GGAAGGGGTGTATTTATTAGATAAAAAACCAATGCCCTTCGGG
C. neoformans   AAAGCCCCGACTTCT--GGAAGGGGTGTATTTATTAGATAAAAAACCAATGGGTTTCGGC
                ***  * ****      ***** * ******************** *****   *  *

C. albicans     TCTTT--GATGATTCATAATAACTTTTCGAATCGCATGGCCTTGTGCTGGCGATGGTTCA 300
C. tropicalis   TCTTT--GATGATTCATAATAACTTTTCGAATCGCATGGCCTTGTGCTGGCGATGGTTCA
S. bayanus      TCTTT--GATGATTCATAATAACTTTTCGAATCGCATGGCCTTGTGCTGGCGATGGTTCA
S. paradoxus    TCTTT--GATGATTCATAATAACTTTTCGAATCGCATGGCCTTGTGCTGGCGATGGTTCA
S. mikatae      TCTTT--GATGATTCATAATAACTTTTCGAATCGCATGGCCTTGTGCTGGCGATGGTTCA
S. cerevisiae   TCTTT--GATGATTCATAATAACTTTTCGAATCGCATGGCCTTGTGCTGGCGATGGTTCA
C. glabrata     TTTTT--GATGATTCATAATAACTTTTCGAATCGCATGGCCTTGTGCTGGCGATGGTTCA
S. pombe        TTTTTTTGGTGAGTCATAATAACTTTTCGAATCGCATGGCCTTGCGCCGGCGATGGTTCA
A. fumigatus    GCTCCTTGGTGAATCATAATAACTTAACGAATCGCATGGCCTTGCGCCGGCGATGGTTCA
C. neoformans   CCTCTATGGTGAATCATAATAACTTCTCGAATCGCATGGCCTTGTGCCGGCGATGCTTCA
                  *    * *** ************  ***************** ** ******* ****
```

```
C. albicans     TTCAAATTTCTGCCCTATCAACTTTCGATGGTAGGATAGTGGCCTACCATGGTTTCAACG 360
C. tropicalis   TTCAAATTTCTGCCCTATCAACTTTCGATGGTAGGATAGTGGCCTACCATGGTTTCAACG
S. bayanus      TTCAAATTTCTGCCCTATCAACTTTCGATGGTAGGATAGTGGCCTACCATGGTTTCAACG
S. paradoxus    TTCAAATTTCTGCCCTATCAACTTTCGATGGTAGGATAGTGGCCTACCATGGTTTCAACG
S. mikatae      TTCAAATTTCTGCCCTATCAACTTTCGATGGTAGGATAGTGGCCTACCATGGTTTCAACG
S. cerevisiae   TTCAAATTTCTGCCCTATCAACTTTCGATGGTAGGATAGTGGCCTACCATGGTTTCAACG
C. glabrata     TTCAAATTTCTGCCCTATCAACTTTCGATGGTAGGATAGTGGCCTACCATGGTTTCAACG
S. pombe        TTCAAATTTCTGCCCTATCAACTTTCGATGGTAGGATAGAGGCCTACCATGGTTTTAACG
A. fumigatus    TTCAAATTTCTGCCCTATCAACTTTCGATGGTAGGATAGTGGCCTACCATGGTGGCAACG
C. neoformans   TTCAAATATCTGCCCTATCAACTTTCGATGGTAGGATAGAGGCCTACCATGGTATCAACG
                ******* ******************************** *************   ****

C. albicans     GGTAACGGGGAATAAGGGTTCGATTCCGGAGAGGGAGCCTGAGAAACGGCTACCACATCC 420
C. tropicalis   GGTAACGGGGAATAAGGGTTCGATTCCGGAGAGGGAGCCTGAGAAACGGCTACCACATCC
S. bayanus      GGTAACGGGGAATAAGGGTTCGATTCCGGAGAGGGAGCCTGAGAAACGGCTACCACATCC
S. paradoxus    GGTAACGGGGAATAAGGGTTCGATTCCGGAGAGGGAGCCTGAGAAACGGCTACCACATCC
S. mikatae      GGTAACGGGGAATAAGGGTTCGATTCCGGAGAGGGAGCCTGAGAAACGGCTACCACATCC
S. cerevisiae   GGTAACGGGGAATAAGGGTTCGATTCCGGAGAGGGAGCCTGAGAAACGGCTACCACATCC
C. glabrata     GGTAACGGGGAATAAGGGTTCGATTCCGGAGAGGGAGCCTGAGAAACGGCTACCACATCC
S. pombe        GGTAACGGGGAATTAGGGTTCGATTCCGGAGAGGGAGCCTGAGAAACGGCTACCACATCC
A. fumigatus    GGTAACGGGGAATTAGGGTTCGATTCCGGAGAGGGAGCCTGAGAAACGGCTACCACATCC
C. neoformans   GGTAACGGGGAATTAGGGTTCGATTCCGGAGAGGGAGCCTGAGAAACGGCTACCACATCC
                ************* ************************************************

C. albicans     AAGGAAGGCAGCAGGCGCGCAAATTACCCAATCCCGACACGGGGAGGTAGTGACAATAAA 480
C. tropicalis   AAGGAAGGCAGCAGGCGCGCAAATTACCCAATCCCGACACGGGGAGGTAGTGACAATAAA
S. bayanus      AAGGAAGGCAGCAGGCGCGCAAATTACCCAATCCTAATTCAGGGAGGTAGTGACAATAAA
S. paradoxus    AAGGAAGGCAGCAGGCGCGCAAATTACCCAATCCTAATTCAGGGAGGTAGTGACAATAAA
S. mikatae      AAGGAAGGCAGCAGGCGCGCAAATTACCCAATCCTAATTCAGGGAGGTAGTGACAATAAA
S. cerevisiae   AAGGAAGGCAGCAGGCGCGCAAATTACCCAATCCTAATTCAGGGAGGTAGTGACAATAAA
C. glabrata     AAGGAAGGCAGCAGGCGCGCAAATTACCCAATCCTGACACAGGGAGGTAGTGACAATAAA
S. pombe        AAGGAAGGCAGCAGGCGCGCAAATTACCCAATCCCGACACGGGGAGGTAGTGACAAGAAA
A. fumigatus    AAGGAAGGCAGCAGGCGCGCAAATTACCCAATCCCGACACGGGGAGGTAGTGACAATAAA
C. neoformans   AAGGAAGGCAGCAGGCGCGCAAATTACCCAATCCCGACACGGGGAGGTAGTGACAATAAA
                ********************************  *  * *************** ***

C. albicans     TAACGATACAGGGCCCTTTTGGGTCTTGTAATTGGAATGAGTACAATGTAAATACCTTAA 540
C. tropicalis   TAACGATACAGGGCCCTTTCGGGTCTTGTAATTGGAATGAGTACAATGTAAATACCTTAA
S. bayanus      TAACGATACAGGGCCCATTCGGGTCTTGTAATTGGAATGAGTACAATGTAAATACCTTAA
S. paradoxus    TAACGATACAGGGCCCATTCGGGTCTTGTAATTGGAATGAGTACAATGTAAATACCTTAA
S. mikatae      TAACGATACAGGGCCCATTCGGGTCTTGTAATTGGAATGAGTACAATGTAAATACCTTAA
S. cerevisiae   TAACGATACAGGGCCCATTCGGGTCTTGTAATTGGAATGAGTACAATGTAAATACCTTAA
C. glabrata     TAACGATACAGGGCCCATTCGGGTCTTGTAATTGGAATGAGTACAATGTAAATACCTTAA
S. pombe        TAACAATGCAGGGCCCTTTCGGGTCTTGTAATTGGAATGAGTACAATGTAAATACCTTAA
A. fumigatus    TACTGATACGGGGCTCTTTTGGGTCTCGTAATTGGAATGAGTACAATCTAAATCCCTTAA
C. neoformans   TAACAATACAGGGCTCTTTTGGGCCTTGTAATTGGAATGAGTACAATTTAAATCCCTTAA
                **   ** * **** * ** *** ** ******************** ***** ******

C. albicans     CGAGGAACAATTGGAGGGCAAGTCTGGTGCCAGCAGCCGCGGTAATTCCAGCTCCAAAAG 600
C. tropicalis   CGAGGAACAATTGGAGGGCAAGTCTGGTGCCAGCAGCCGCGGTAATTCCAGCTCCAAAAG
S. bayanus      CGAGGAACAATTGGAGGGCAAGTCTGGTGCCAGCAGCCGCGGTAATTCCAGCTCCAATAG
S. paradoxus    CGAGGAACAATTGGAGGGCAAGTCTGGTGCCAGCAGCCGCGGTAATTCCAGCTCCAATAG
S. mikatae      CGAGGAACAATTGGAGGGCAAGTCTGGTGCCAGCAGCCGCGGTAATTCCAGCTCCAATAG
S. cerevisiae   CGAGGAACAATTGGAGGGCAAGTCTGGTGCCAGCAGCCGCGGTAATTCCAGCTCCAATAG
C. glabrata     CGAGGAACAATTGGAGGGCAAGTCTGGTGCCAGCAGCCGCGNTAATTCCAGCTCCAATAG
S. pombe        CGAGGAACAATTGGAGGGCAAGTCTGGTGCCAGCAGCCGCGGTAATTCCAGCTCCAATAG
A. fumigatus    CGAGGAACAATTGGAGGGCAAGTCTGGTGCCAGCAGCCGCGGTAATTCCAGCTCCAATAG
C. neoformans   CGAGGAACAACTGGAGGGCAAGTCTGGTGCCAGCNGCCGCGGTAATTCCAGCTCCAGTAG
                ********** ********************** ****** **************  **

C. albicans     CGTATATTAAAGTTGTTGCAGTTAAAAAGCTCGTAGTTGAACCTTGGGCTTGGCTGGCCG 660
C. tropicalis   CGTATATTAAAGTTGTTGCAGTTAAAAAGCTCGTAGTTGAACCTTGGGCTTGGTTGGCCG
S. bayanus      CGTATATTAAAGTTGTTGCAGTTAAAAAGCTCGTAGTTGAACTTTGGGCCCGGTTGGCCG
S. paradoxus    CGTATATTAAAGTTGTTGCAGTTAAAAAGCTCGTAGTTGAACTTTGGGCCCGGTTGGCCG
S. mikatae      CGTATATTAAAGTTGTTGCAGTTAAAAAGCTCGTAGTTGAACTTTGGGCTCGGTTGGCCG
S. cerevisiae   CGTATATTAAAGTTGTTGCAGTTAAAAAGCTCGTAGTTGAACTTTGGGCCCGGTTGGCCG
C. glabrata     CGTATATTAAAGTTGTTGCAGTTAAAAAGCTCGTAGTTGAACTTTGGGCCTGGGTNGCCG
S. pombe        CGTATATTAAAGTTGTTGCAGTTAAAAAGCTCGTAGTTGAACTTTGAGCCTGGTCGACTG
A. fumigatus    CGTATATTAAAGTTGTTGCAGTTAAAAAGCTCGTAGTTGAACCTTGGGTCTGGCTGGCCG
C. neoformans   CGTATATTAAAGTTGTTGCAGTTAAAAAGCTCGTAGTCGAACTTCAGGTCTGGCGAGGCG
                ************************************ **** *   *   **      *
```

```
C. albicans     GTCCATCTTTTTG-ATGCGTACTGGA--CCCAGCCGAGCCTTT--CCTTCTGGGTAGC-- 720
C. tropicalis   GTCCATCTTTCTG-ATGCGTACTGGA--CCCAACCGAGCCTTT--CCTTCTGGCTAGC--
S. bayanus      GTCCGATTTTTT---CGTGTACTGGATTTCCAACGGGGCCTTT--CCTTCTGGCTAAC--
S. paradoxus    GTCCGATTTTTT---CGTGTACTGGATTTCCAACGGGGCCTTT--CCTTCTGGCTAAC--
S. mikatae      GTCCGATTTTTT---CGTGTACTGGATTTCCAACGGGGCCTTT--CCTTCTGGCTAAC--
S. cerevisiae   GTCCGATTTTTT---CGTGTACTGGATTTCCAACGGGGCCTTT--CCTTCTGGCTAAC--
C. glabrata     GTCCGATTTTTT---CGTGTACTGGAATGC-ACCCGGGCCTTT--CCTTCTGGCTAAC--
S. pombe        GTCCGCCGCAAGGCGTGTTTACTGGT-CATGACCGGGGTCGTTAACCTTCTGGCAAACTA
A. fumigatus    GTCCGCCTCACCG--CGAGTACTGGT---CCGGCTGGACCTTT--CCTTCTGGGGAAC--
C. neoformans   GTCCTCCTCACGG--AGTGCACTG----TCTTGCTGGACCTTA--CCTCCTGGTGGTCC-
                ****            *   ****         * *   * *   *** ****    *

C. albicans     CA-----------------TTTATGGCGAACCAGGACTTTTACTTTGAAAAAATTAGAGT 780
C. tropicalis   CT-----------------TTT--GGCGAACCAGGACTTTTACTTTGAAAAAATTAGAGT
S. bayanus      CTTGAGTCCTTGT---GGCTCT-TGGCGAACCAGGACTTTTACTTTGAAAAAATTAGAGT
S. paradoxus    CTTGAGTCCTTGT---GGCTCT-TGGCGAACCAGGACTTTTACTTTGAAAAAATTAGAGT
S. mikatae      CTTGAGTCCTTGT---GGCTCT-TGGCGAACCAGGACTTTTACTTTGAAAAAATTAGAGT
S. cerevisiae   CTTGAGTCCTTGT---GGCTCT-TGGCGAACCAGGACTTTTACTTTGAAAAAATTAGAGT
C. glabrata     CCCAAGTCCTTGT---GGCTTGGCGGCGAACCAGGACTTTTACTTTGAAAAAATTAGAGT
S. pombe        CTCATGTTCTTTATTGAGCGTGGTAGGGAACCAGGACTTTTACCTTGAAAAAATTAGAGT
A. fumigatus    CTCATGGCCTTCACT-GGCTGTGGGGGGAACCAGGACTTTTACTGTGAAAAAATTAGAGT
C. neoformans   TGTATGCTCTTTACTGGGTGTGCAGGGGAACCAGGAATTTTACCTTGAAAAAATTAGAGT
                                        * ********* ******  ***************

C. albicans     GTTCAAAGCAGGC--CT-TTGCTCGAATATATTAGCATGGAATAATAGAATAGGACGTTA 840
C. tropicalis   GTTCAAAGCAGGC--CT-TTGCTCGAATATATTAGCATGGAATAATAGAATAGGACGTTA
S. bayanus      GTTCAAAGCAGGC--GTATTGCTCGAATATATTAGCATGGAATAATAGAATAGGACGTT-
S. paradoxus    GTTCAAAGCAGGC--GTATTGCTCGAATATATTAGCATGGAATAATAGAATAGGACGTT-
S. mikatae      GTTCAAAGCAGGC--GTATTGCTCGAATATATTAGCATGGAATAATAGAATAGGACGTT-
S. cerevisiae   GTTCAAAGCAGGC--GTATTGCTCGAATATATTAGCATGGAATAATAGAATAGGACGTT-
C. glabrata     GTTCAAAGCAGGC--GTATTGCTCGAATATATTAGCATGGAATAATGGAATAGGACGTT-
S. pombe        GTTCAAAGCAGGCAAGTTTTGCTCGAATACATTAGCATGGAATAATAAAATAGGACGTG-
A. fumigatus    GTTCAAAGCAGGC---CTTTGCTCGAATACATTAGCATGGAATAATAGAATAGGACGTG-
C. neoformans   GTTCAAAGCAGGC---AATCGCCCGAATACATTAGCATGGAATAATAGAATAGGACGTG-
                **************     * ** ****** ****************  **********

C. albicans     TGGTTCTATTTTGTTGGTTTCTAGGACCATCGTAATGATTAATAGGGACGGTCGGGGGTA 900
C. tropicalis   TGGTTCTATTTTGTTGGTTTCTAGGACCATCGTAATGATTAATAGGGACGGTCGGGGGTA
S. bayanus      TGGTTCTATTTTGTTGGTTTCTAGGACCATCGTAATGATTAATAGGGACGGTCGGGGGCA
S. paradoxus    TGGTTCTATTTTGTTGGTTTCTAGGACCATCGTAATGATTAATAGGGACGGTCGGGGGCA
S. mikatae      TGGTTCTATTTTGTTGGTTTCTAGGACCATCGTAATGATTAATAGGGACGGTCGGGGGCA
S. cerevisiae   TGGTTCTATTTTGTTGGTTTCTAGGACCATCGTAATGATTAATAGGGACGGTCGGGGGCA
C. glabrata     TGGTTCTATTTTGTTGGTTTCTAGGACCATCGTAATGATTAATAGGGACGGTCGGGGGCA
S. pombe        TGGTTCTATTTTGTTGGTTTCTAGGACCGCCGTAATGATTAATAGGGATAGTCGGGGGCA
A. fumigatus    CGGTTCTATTTTGTTGGTTTCTAGGACCGCCGTAATGATTAATAGGGATAGTCGGGGGCG
C. neoformans   CGGTTCTATTTTGTTGGTTTCTAGGATCGCCGTAATGATTAATAGGGACGGTCGGGGGCA
                 ************************* *  *****************  ********

C. albicans     TCAGTATTCAGTTGTCAGAGGTGAAATTCTTGGATTTACTGAAGACTAACTACTGCGAAA 960
C. tropicalis   TCAGTATTCAGTTGTCAGAGGTGAAATTCTTGGATTTACTGAATACTAACTACTGCGAAA
S. bayanus      TCAGTATTCAATTGTCAGAGGTGAAATTCTTGGATTTATTGAAGACTAACTACTGCGAAA
S. paradoxus    TCAGTATTCAATTGTCAGAGGTGAAATTCTTGGATTTATTGAAGACTAACTACTGCGAAA
S. mikatae      TCAGTATTCAATTGTCAGAGGTGAAATTCTTGGATTTATTGAAGACTAACTACTGCGAAA
S. cerevisiae   TCAGTATTCAATTGTCAGAGGTGAAATTCTTGGATTTATTGAAGACTAACTACTGCGAAA
C. glabrata     TCAGTATTCAATTGTCAGAGGTGAAATTCTTGGATTTATTGAAGACTAACTACTGCGAAA
S. pombe        TTCGTATTCAATTGTCAGAGGTGAAATTCTTGGATTTATTGAAGACGAACTACTGCGAAA
A. fumigatus    TCAGTATTCAGCTGTCAGAGGTGAAATTCTTGGATTTGCTGAAGACTAACTACTGCGAAA
C. neoformans   TTGGTATTCCGTTGCTAGAGGTGAAATTCTTAGATTGACGGAAGACCAACAACTGCGAAA
                *  ******   **  *************** ****    *** ** *** *********

C. albicans     GCATTTACCAAGGACGTTTTCATTAATCAAG-AACGAAAGTTAGGGGATCGAAGATGATC 1020
C. tropicalis   GCATTTACCAAGGACGTTTTCATTAATCAAG-AACGAAAGTTAGGGGATCGAAGATGATC
S. bayanus      GCATTTGCCAAGGACGTTTTCATTAATCAAG-AACGAAAGTTAGGGGATCGAAGATGATC
S. paradoxus    GCATTTGCCAAGGACGTTTTCATTAATCAAG-AACGAAAGTTAGGGGATCGAAGATGATC
S. mikatae      GCATTTGCCAAGGACGTTTTCATTAATCAAG-AACGAAAGTTAGGGGATCGAAGATGATC
S. cerevisiae   GCATTTGCCAAGGACGTTTTCATTAATCAAG-AACGAAAGTTAGGGGATCGAAGATGATC
C. glabrata     GCATTTGCCAAGGACGTTTTCATTAATCAAG-AACGAAAGTTAGGGGATCGAAGATGATC
S. pombe        GCATTTGCCAAGGATGTTTTCATTAATCAAG-AACGAAAGTTAGGGGATCGAAGACGATC
A. fumigatus    GCATTCGCCAAGGATGTTTTCATTAATCAGGGAACGAAAGTTAGGGGATCGAAGACGATC
C. neoformans   GCATTTGCCAAGGACGTTTTCATTGATCAAG-AACGAAGGTTAGGGGATCAAAAACGATT
                *****  ******* ********* **** * ****** ********** ** * ***
```

```
C. albicans     AGATACCGTCGTAGTCTTAACCATAAACTATGCCGACTAGGGATCGGTTGTTGTTCT-TT 1080
C. tropicalis   AGATACCGTCGTAGTCTTAACCATAAACTATGCCGACTAGGGATCGGTTGTTGTTCT-TT
S. bayanus      AGATACCGTCGTAGTCTTAACCATAAACTATGCCGACTAGGGATCGGGTGGTGTTTT-TT
S. paradoxus    AGATACCGTCGTAGTCTTAACCATAAACTATGCCGACTAGGGATCGGGTGGTGTTTT-TT
S. mikatae      AGATACCGTCGTAGTCTTAACCATAAACTATGCCGACTAGGGATCGGGTGGTGTTTT-TT
S. cerevisiae   AGATACCGTCGTAGTCTTAACCATAAACTATGCCGACTAGGGATCGGGTGGTGTTTT-TT
C. glabrata     AGATACCGTCGTAGTCTTAACCATAAACTATGCCGACTAGGGATCGGGTGGTGTTTT-TT
S. pombe        AGATACCGTCGTAGTCTTAACCATAAACTATGCCGACTAGGGATCGGGCAATGTTTCATT
A. fumigatus    AGATACCGTCGTAGTCTTAACCATAAACTATGCCGACTAGGGATCGGGCGGTGTTTC-TA
C. neoformans   AGATACCGTTGTAGTCTTAACAGTAAACGATGCCGACTAGGGATCGGCCCACGTCAA--T
                ********* ***********  ***** ******************     **

C. albicans     TATTGACGCAATCGGCACCTTACGAGAAATCAAAGTCTTTGGGTTCTGGGGGGAGTATGG 1140
C. tropicalis   TATTGACGCAATCGGCACCTTACGAGAAATCAAAGTCTTTGGGTTCTGGGGGGAGTATGG
S. bayanus      TAATGACCCACTCGGCACCTTACGAGAAATCAAAGTCTTTGGGTTCTGGGGGGAGTATGG
S. paradoxus    TAATGACCCACTCGGCACCTTACGAGAAATCAAAGTCTTTGGGTTCTGGGGGGAGTATGG
S. mikatae      TAATGACCCACTCGGCACCTTACGAGAAATCAAAGTCTTTGGGTTCTGGGGGGAGTATGG
S. cerevisiae   TAATGACCCACTCGGCACCTTACGAGAAATCAAAGTCTTTGGGTTCTGGGGGGAGTATGG
C. glabrata     TAGTGACCCACTCGGCACCTTACGAGAAATCAAAGTCTTTGGGTTCTGGGGGGAGTATGG
S. pombe        TATCGACTTGCTCGGCACCTTACGAGAAATCAAAGTCTTTGGGTTCCGGGGGGAGTATGG
A. fumigatus    TGATGACCCGCTCGGCACCTTACGAGAAATCAAAGTTTTTGGGTTCTGGGGGGAGTATGG
C. neoformans   CTCTGACTGGGTCGGCACCTTACGAGAAATCAAAGTCTTTGGGTTCTGGGGGGAGTATGG
                    ***    ************************ ********* *************

C. albicans     TCGCAAGGCTGAAACTTAAAGGAATTGACGGAAGGGCACCACCA-GGAGTGGAGCCTGCG 1200
C. tropicalis   TCGCAAGGCTGAAACTTAAAGGAATTGACGGAAGGGCACCACCA-GGAGTGGAGCCTGCG
S. bayanus      TCGCAAGGCTGAAACTTAAAGGAATTGACGGAAGGGCACCACCA-GGAGTGGAGCCTGCG
S. paradoxus    TCGCAAGGCTGAAACTTAAAGGAATTGACGGAAGGGCACCACCA-GGAGTGGAGCCTGCG
S. mikatae      TCGCAAGGCTGAAACTTAAAGGAATTGACGGAAGGGCACCACCA-GGAGTGGAGCCTGCG
S. cerevisiae   TCGCAAGGCTGAAACTTAAAGGAATTGACGGAAGGGCACCACCA-GGAGTGGAGCCTGCG
C. glabrata     TCGCAAGGCTGAAACTTAAAGGAATTGACGGAAGGGCACCACCA-GGAGTGGAGCCTGCG
S. pombe        TCGCAAGGCTGAAACTTAAAGGAATTGACGGAAGGGCACCACAATGGAGTGGAGCCTGCG
A. fumigatus    TCGCAAGGCTGAAACTTAAAGAAATTGACGGAAGGGCACCACAA-GGCGTGGAGCCTGCG
C. neoformans   TCGCAAGGCTGAAACTTAAAGGAATTGACGGAAGGGCACCACCA-GGTGTGGAGCCTGCG
                ********************** ******************** * ** ************

C. albicans     GCTTAATTTGACTCAACACGGGGAAACTCACCAGGTCCAGACACAATAAGGATTGACAGA 1260
C. tropicalis   GCTTAATTTGACTCAACACGGGGAAACTCACCAGGTCCAGACACAATAAGGATTGACAGA
S. bayanus      GCTTAATTTGACTCAACACGGGGAAACTCACCAGGTCCAGACACAATAAGGATTGACAGA
S. paradoxus    GCTTAATTTGACTCAACACGGGGAAACTCACCAGGTCCAGACACAATAAGGATTGACAGA
S. mikatae      GCTTAATTTGACTCAACACGGGGAAACTCACCAGGTCCAGACACAATAAGGATTGACAGA
S. cerevisiae   GCTTAATTTGACTCAACACGGGGAAACTCACCAGGTCCAGACACAATAAGGATTGACAGA
C. glabrata     NCTTAATTTGACTCAACACGGGGAAACTCACCAGGTCCAGACACAATAAGGATTGACAGA
S. pombe        GCTTAATTTGACTCAACACGGGGAAACTCACCAGGTCCAGACATAGTAAGGATTGACAGA
A. fumigatus    GCTTAATTTGACTCAACACGGGGAAACTCACCAGGTCCAGACAAAATAAGGATTGACAGA
C. neoformans   GCTTAATTTGACTCAACACGGGGAAACTCACCAGGTCCAGACATAGTGAGGATTGACAGA
                 ****************************************** * * ************

C. albicans     TTGAGAGCTCTTTCTTGATTTTGTGGGTGGTGGTGCATGGCCGTTCTTAGTTGGTGGAGT 1320
C. tropicalis   TTGAGAGCTCTTTCTTGATTTTGTGGGTGGTGGTGCATGGCCGTTCTTAGTTGGTGGAGT
S. bayanus      TTGAGAGCTCTTTCTTGATTTTGTGGGTGGTGGTGCATGGCCGTTCTTAGTTGGTGGAGT
S. paradoxus    TTGAGAGCTCTTTCTTGATTTTGTGGGTGGTGGTGCATGGCCGTTCTTAGTTGGTGGAGT
S. mikatae      TTGAGAGCTCTTTCTTGATTTTGTGGGTGGTGGTGCATGGCCGTTCTTAGTTGGTGGAGT
S. cerevisiae   TTGAGAGCTCTTTCTTGATTTTGTGGGTGGTGGTGCATGGCCGTTCTTAGTTGGTGGAGT
C. glabrata     TTGAGAGCTCTTTCTTGATTTTGTGGGTGGTGGTGCATGGCCGTTCTTAGTTGGTGGAGT
S. pombe        TTGAGAGCTCTTTCTTGATTCTATGGGTGGTGGTGCATGGCCGTTCTTAGTTGGTGGAGT
A. fumigatus    TTGAGAGCTCTTTCTTGATCTTTTGGATGGTGGTGCATGGCCGTTCTTAGTTGGTGGAGT
C. neoformans   TTGATAGCTCTTTCTTGATTCTATGGGTGGTGGTGCATGGCCGTTCTTAGTTGGTGGAGT
                **** ***************  * *** ********************************

C. albicans     GATTTGTCTGCTTAATTGCGATAACGAACGAGACCTTAACCTACTAAATAGTGCTGCTAG 1380
C. tropicalis   GATTTGTCTGCTTAATTGCGATAACGAACGAGACCTTAACCTACTAAATAGTGCTGCTAG
S. bayanus      GATTTGTCTGCTTAATTGCGATAACGAACGAGACCTTAACCTACTAAATAGTGGTGCTAG
S. paradoxus    GATTTGTCTGCTTAATTGCGATAACGAACGAGACCTTAACCTACTAAATAGTGGTGCTAG
S. mikatae      GATTTGTCTGCTTAATTGCGATAACGAACGAGACCTTAACCTACTAAATAGTGGTGCTAG
S. cerevisiae   GATTTGTCTGCTTAATTGCGATAACGAACGAGACCTTAACCTACTAAATAGTGGTGCTAG
C. glabrata     GATTTGTCTGCTTAATTGCGATAACGAACGAGACCTTAACCTACTAAATAGTGGTGCTAG
S. pombe        GATTTGTCTGCTTAATTGCGATAACGAACGAGACCTTAACCTGCTAAATAGCTGGATCAG
A. fumigatus    GATTTGTCTGCTTAATTGCGATAACGAACGAGACCTCGGCCC-TTAAATAGCCCGGTCCG
C. neoformans   GATTTGTCTGGTTAATTCCGATAACGAACGAGACCTTAACCTGCTAAATAGTCAGGCCGG
                ********** ****** *****************   **   *******        *
```

```
C. albicans     C-ATTT--GCTGGTATAGTCACTTCTTAGAGGGACTATCGACTTCAAGTCGATGGAAGTT 1440
C. tropicalis   C-ATTT--GCTGGTATAGTCACTTCTTAGAGGGACTATCGATTTCAAGTCGATGGAAGTT
S. bayanus      C-ATTT--GCTGGT-TATCCACTTCTTAGAGGGACTATCGGTTTCAAGCCGATGGAAGTT
S. paradoxus    C-ATTT--GCTGGT-TATCCACTTCTTAGAGGGACTATCGGTTTCAAGCCGATGGAAGTT
S. mikatae      C-ATTT--GCTGGT-TATCCACTTCTTAGAGGGACTATCGGTTTCAAGCCGATGGAAGTT
S. cerevisiae   C-ATTT--GCTGGT-TATCCACTTCTTAGAGGGACTATCGGTTTCAAGCCGATGGAAGTT
C. glabrata     C-ATTT--GCTGGT-TGTCCACTTCTTAGAGGGACTATCGGTTTCAAGCCGATGGAAGTT
S. pombe        CCATTTTGGCTGAT-CATTAGCTTCTTAGAGGGACTATTGGCATAAAGCCAATGGAAGTT
A. fumigatus    C-ATTT--GCGGGC-CGCTGGCTTCTTAGGGGGACTATCGGC-TCAAGCCGATGGAAGTG
C. neoformans   C--TTT-GGCTGGTCGTTTGACTTCTTAGAGGGACTGTCGGCGTCTAGTCGACGGAAGTT
                *  ***  ** *         ******** ****** * *   *  ** * * ******

C. albicans     TGAGGCAATAACAGGTCTGTGATGCCCTTAGACGTTCTGGGCCGCACGCGCGCTACACTG 1500
C. tropicalis   TGAGGCAATAACAGGTCTGTGATGCCCTTAGACGTTCTGGGCCGCACGCGCGCTACACTG
S. bayanus      TGAGGCAATAACAGGTCTGTGATGCCCTTAGACGTTCTGGGCCGCACGCGCGCTACACTG
S. paradoxus    TGAGGCAATAACAGGTCTGTGATGCCCTTAGACGTTCTGGGCCGCACGCGCGCTACACTG
S. mikatae      TGAGGCAATAACAGGTCTGTGATGCCCTTAGACGTTCTGGGCCGCACGCGCGCTACACTG
S. cerevisiae   TGAGGCAATAACAGGTCTGTGATGCCCTTAGACGTTCTGGGCCGCACGCGCGCTACACTG
C. glabrata     TGAGGCAATAACAGGTCTGTGATGCCCTTAGACGTTCTGGGCCGCACGCGCGCTACACTG
S. pombe        TGAGGCAATAACAGGTCTGTGATGCCCTTAGATGTTCTGGGCCGCACGCGCGCTACACTG
A. fumigatus    CGCGGCAATAACAGGTCTGTGATGCCCTTAGATGTTCTGGGCCGCACGCGCGCTACACTG
C. neoformans   TGAGGCAATAACAGGTCTGTGATGCCCTTAGATGTTCTGGGCCGCACGCGCGCTACACTG
                 * ***************************** *************************

C. albicans     ACGGAGCCAGCGAGTATAAG--------------------CCTTGGCCGAGAGGTCTGGG 1560
C. tropicalis   ACGGAGCCAGCGAGTATAAA--------------------CCTTGGCCGAGAGGCCTGGG
S. bayanus      ACGGAGCCAGCGAGTCTAA---------------------CCTTGGCCGAGAGGTCTTGG
S. paradoxus    ACGGAGCCAGCGAGTCTAA---------------------CCTTGGCCGAGAGGTCTTGG
S. mikatae      ACGGAGCCAGCGAGTCTAA---------------------CCTTGGCCGAGAGGTCTTGG
S. cerevisiae   ACGGAGCCAGCGAGTCTAA---------------------CCTTGGCCGAGAGGTCTTGG
C. glabrata     ACGGAGCCAGCGAGTCTAA---------------------CCTTGGCCGAGAGGTCTTGG
S. pombe        ACGGAGCCAACGAGTTGAAAAAAATCTTTTGATTTTTTATCCTTGGCCGGAAGGTCTGGG
A. fumigatus    ACAGGGCCAGCGAGTACATCA-------------------CCTTGGCCGAGAGGTCTGGG
C. neoformans   ACTGAGCCAGCGAGTCTTACCG-----------------CCTTGGCCGAGAGGCCTGGG
                ** * **** *****                        *********  *** ** **

C. albicans     AAATCTTGTGAAACTCCGTCGTGCTGGGGATAGAGCATTGTAATTGTTGCTCTTCAACGA 1620
C. tropicalis   AAATCTTGTGAAACTCCGTCGTGCTGGGGATAGAGCATTGTAATTGTTGCTCTTCAACGA
S. bayanus      TAATCTTGTGAAACTCCGTCGTGCTGGGGATAGAGCATTGTAATTATTGCTCTTCAACGA
S. paradoxus    TAATCTTGTGAAACTCCGTCGTGCTGGGGATAGAGCATTGTAATTATTGCTCTTCAACGA
S. mikatae      TAATCTTGTGAAACTCCGTCGTGCTGGGGATAGAGCATTGTAATTATTGCTCTTCAACGA
S. cerevisiae   TAATCTTGTGAAACTCCGTCGTGCTGGGGATAGAGCATTGTAATTATTGCTCTTCAACGA
C. glabrata     TAATCTTGTGAAACTCCGTCGTGCTGGGGATAGAGCATTGTAATTATTGCTCTTCAACGA
S. pombe        TAATCTTGTTAAACTCCGTCGTGCTGGGGATAGAGCATTGCAATTATTGCTCTTCAACGA
A. fumigatus    TAATCTTGTTAAACCCTGTCGTGCTGGGGATAGAGCATTGCAATTATTGCTCTTCAACGA
C. neoformans   TAATCTTGTGAAACTCAGTCGTGCTGGGGATAGAGCATTGCAATTATTGCTCTTCAACGA
                 ******** **** * ********************** **** **************

C. albicans     GGAATTCCTAGTAAGCGCAAGTCATCAGCTTGCGTTGATTACGTCCCTGCCCTTTGTACA 1680
C. tropicalis   GGAATTCCTAGTAAGCGCAAGTCATCAGCTTGCGTTGATTACGTCCCTGCCCTTTGTACA
S. bayanus      GGAATTCCTAGTAAGCGCAAGTCATCAGCTTGCGTTGATTACGTCCCTGCCCTTTGTACA
S. paradoxus    GGAATTCCTAGTAAGCGCAAGTCATCAGCTTGCGTTGATTACGTCCCTGCCCTTTGTACA
S. mikatae      GGAATTCCTAGTAAGCGCAAGTCATCAGCTTGCGTTGATTACGTCCCTGCCCTTTGTACA
S. cerevisiae   GGAATTCCTAGTAAGCGCAAGTCATCAGCTTGCGTTGATTACGTCCCTGCCCTTTGTACA
C. glabrata     GGAATTCCTAGTAAGCGCAAGTCATCAGCTTGCGTTGATTACGTCCCTGCCCTTTGTACA
S. pombe        GGAATTCCTAGTAAGCGCAAGTCATCAGCTTGCGTTGAATACGTCCCTGCCCTTTGTACA
A. fumigatus    GGAATGCCTAGTAGGCACGAGTCATCAGCTCGTGCCGATTACGTCCCTGCCCTTTGTACA
C. neoformans   GGAATACCTAGTAAGCGTGAGTCACCAGCTCGCGTTGATTACGTCCCTGCCCTTTGTACA
                ***** ******* **   ***** ***** * *  ** ********************

C. albicans     CACCGCCCGTCGCTACTACCGATTGAATGGCTTAGTGAGGCCTCCGGATTGGTTTAGGAA 1740
C. tropicalis   CACCGCCCGTCGCTACTACCGATTGAATGGCTTAGTGAGGCTTCCGGATTGGTTTAGGAA
S. bayanus      CACCGCCCGTCGCTAGTACCGATTGAATGGCTTAGTGAGGCCTCAGGATCTGCTTAGAGA
S. paradoxus    CACCGCCCGTCGCTAGTACCGATTGAATGGCTTAGTGAGGCCTCAGGATCTGCTTAGAGA
S. mikatae      CACCGCCCGTCGCTAGTACCGATTGAATGGCTTAGTGAGGCCTCAGGATCTGCTTAGAGA
S. cerevisiae   CACCGCCCGTCGCTAGTACCGATTGAATGGCTTAGTGAGGCCTCAGGATCTGCTTAGAGA
C. glabrata     CACCGCCCGTCGCTAGTACCGATTGAATGGCTTAGTGAGGCCTCAGGATCTGCTTAGAAG
S. pombe        CACCGCCCGTCGCTACTACCGATTGAATGGCTTAGTGAGGCCTCTGGATTGGCTTGTTTC
A. fumigatus    CACCGCCCGTCGCTACTACCGATTGAATGGCTCGGTGAGGCCTTCGGACTGGCTCAGGGG
C. neoformans   CACCGCCCGTCGCTACTACCGATTGAATGGCTTAGTGAGATCTCNGGATTGGCGTTGGGG
                *************** ***************  *****   *  ***   *
```

```
C. albicans     AGGGGGCAACCTCATTC-TGGAACCGAGAAGCTGGTCAAACTTGGTCATTTAGAGGAAGT 1800
C. tropicalis   AGGGGGCAACTCCATTC-TGGAACCGAGAAGCTAGTCAAACTTGGTCATTTAGAGGAAGT
S. bayanus      AGGGGGCAACTCCATCT-CAGAGCGGAGAATTTGGACAAACTTGGTCATTTAGAGGAACT
S. paradoxus    AGGGGGCAACTCCATCT-CAGAGCGGAGAATTTGGACAAACTTGGTCATTTAGAGGAACT
S. mikatae      AGGGGGCAACTCCATCT-CAGAGCGGAGAATTTGGACAAACTTGGTCATTTAGAGGAACT
S. cerevisiae   AGGGGGCAACTCCATCT-CAGAGCGGAGAATTTGGACAAACTTGGTCATTTAGAGGAACT
C. glabrata     AGGGGGCGACTCCACTT-CAGAGCGGAGAATCTGGTCAAACTTGGTCATTTAGAGGAACT
S. pombe        TGCTGGCAACGGCGGAAACATTGCCGAGAAGTTGGACAAACTTGGTCATTTAGAGGAAGT
A. fumigatus    AGTTGGCAACGACTCCC-CAGAGCCGGAAAGTTGGTCAAACCCGGTCATTTAGAGGAAGT
C. neoformans   AGCCGGCAACGGCACCC-CTTGGCNGAGAAGCTGATCAAACTTGGTCATTTAGAGGAAGT
                 *  *** **  *          * *  **  *   *****  *************** *

C. albicans     AAAAGTCGTAACAAGGTTTCCGTAGGTGAACCTGCGGAAGGATCATTA 1848
C. tropicalis   AAAAGTCGTAACAAGGTTTCCGTAGGCGAACCTGCGGAAGGATCATTA
S. bayanus      AAAAGTCGTAACAAGGTTTCCGTAGGTGAACCTGCGGAAGGATCATTA
S. paradoxus    AAAAGTCGTAACAAGGTTTCCGTAGGTGAACCTGCGGAAGGATCATTA
S. mikatae      AAAAGTCGTAACAAGGTTTCCGTAGGTGAACCTGCGGAAGGATCATTA
S. cerevisiae   AAAAGTCGTAACAAGGTTTCCGTAGGTGAACCTGCGGAAGGATCATTA
C. glabrata     AAAAGTCGTAACAAGGTTTCCGTAGGTGAACNTGCGGAAGGATCATTA
S. pombe        AAAAGTCGTAACAAGGTTTCCGTAGGTGAACCTGCGGAAGGATCATTA
A. fumigatus    AAAAGTCGTAACAAGGTTTCCGTAGGTGAACCTGCNGAAGGATCATTA
C. neoformans   AAAAGTCGTAACAAGGTTTCCGTAGGTGAACCTGCNGAAGGATCA---
                ************************** **** *** *********
```

Anexo 3

Alinhamento múltiplo de sequências do gene de rRNA 5.8S das espécies *Aspergillus fumigatus*, *Candida albicans*, *Candida glabrata*, *Candida tropicalis*, *Cryptococcus neoformans*, *Saccharomyces bayanus*, *Saccharomyces cerevisiae*, *Saccharomyces mikatae*, *Saccharomyces paradoxus* e *Schizosaccharomyces pombe* realizado pelo algoritmo Clustal através da interface gráfica ClustalX (versão 1.81).

```
C. albicans     AAACTTTCAACAACGGATCTCTTGGTTCTCGCATCGATGAAGAACGCAGCGAAATGCGAT 60
C. tropicalis   AAACTTTCAACAACGGATCTCTTGGTTCTCGCATCGATGAAGAACGCAGCGAAATGCGAT
S. cerevisiae   AAACTTTCAACAACGGATCTCTTGGTTCTCGCATCGATGAAGAACGCAGCGAAATGCGAT
S. paradoxus    AAACTTTCAACAACGGATCTCTTGGTTCTCGCATCGATGAAGAACGCAGCGAAATGCGAT
S. bayanus      AAACTTTCAACAACGGATCTCTTGGTTCTCGCATCGATGAAGAACGCAGCGAAATGCGAT
S. mikatae      AAACTTTCAACAACGGATCTCTTGGTTCTCGCATCGATGAAGAACGCAGCGAAATGCGAT
A. fumigatus    -AACTTTCAACAACGGATCTCTTGGTTCCGGCATCGATGAAGAACGCAGCGAAATGCGAT
C. glabrata     AAACTTTCAACAATGGATCTCTTGGTTCTCGCATCGATGAAGAACGCAGCGAAATGCGAT
S. pombe        AAACTTTCAGCAACGGATCTCTTGGCTCTCGCATCGATGAAGAACGCAGCGAAATGCGAT
C. neoformans   AAACTTTCAACAACGGATCTCTTGGCTTCCACATCGATGAAGAACGCAGCGAAATGCGAT
                 ******** *** *********** *    ****************************

C. albicans     ACGTAATATGAATTGCAGATATTCGTGAATCATCGAATCTTTGAACGCACATTGCGCCCT 120
C. tropicalis   ACGTAATATGAATTGCAGATATTCGTGAATCATCGAATCTTTGAACGCACATTGCGCCCT
S. cerevisiae   ACGTAATGTGAATTGCAGAATTCCGTGAATCATCGAATCTTTGAACGCACATTGCGCCCC
S. paradoxus    ACGTAATGTGAATTGCAGAATTCCGTGAATCATCGAATCTTTGAACGCACATTGCGCCCC
S. bayanus      ACGTAATGTGAATTGCAGAATTCCGTGAATCATCGAATCTTTGAACGCACATTGCGCCCC
S. mikatae      ACGTAATGTGAATTGCAGAATTCCGTGAATCATCGAATCTTTGAACGCACATTGCGCCCC
A. fumigatus    AAGTAATGTGAATTGCAGAATTCAGTGAATCATCGAGTCTTTGAACGCACATTGCGCCCC
C. glabrata     ACGTAATGTGAATTGCAGAATTCCGTGAATCATCGAATCTTTGAACGCACATTGCGCCCT
S. pombe        ACGTAATGTGAATTGCAGAATTCCGTGAATCATCGAATCTTTGAACGCACATTGCGCCTT
C. neoformans   AAGTAATGTGAATTGCAGAATTCAGTGAATCATCGANTCTTTGAACGCAACTTGCGCCCT
                * ***** ***********  *  ************ ************  *******

C. albicans     CTGGTATTCCGGAGGGCATGCCTGTTTGAGCGTCGTTTC------- 166
C. tropicalis   TTGGTATTCCAAAGGGCATGCCTGTTTGAGCGTCATT---------
S. cerevisiae   TTGGTATTCCAGGGGGCATGCCTGTTTGAGCGTCATTT--------
S. paradoxus    TTGGTATTCCAGGGGGCATGCCTGTTTGAGCGTCATTT--------
S. bayanus      TTGGTATTCCAGGGGGCATGCCTGTTTGAGCGTCATTT--------
S. mikatae      TTGGTATTCCAGGGGGCATGCCTGTTTGAGCGTCATT---------
A. fumigatus    CTGGTATTCCGGGGGGCATGCCTGTCCGAGCGTCATT---------
C. glabrata     CTGGTATTCCGGGGGGCATGCCTGTTTGAGCGTCATTT--------
S. pombe        TGGGTTCTACCAAAGGCATGCCTGTTTGAGTGTCATT---------
C. neoformans   TTGGTATTCCGAAGGGCATGCCTGTTTGAGAGTCATGAAAATCTCA
                  ***  * *    ***********  *** *** *
```

Anexo 4

Secção do alinhamento múltiplo de sequências do gene de rRNA 28S, entre as posições 1 e 900, das espécies *Aspergillus fumigatus*, *Candida albicans*, *Candida glabrata*, *Candida tropicalis*, *Cryptococcus neoformans*, *Saccharomyces bayanus*, *Saccharomyces cerevisiae*, *Saccharomyces mikatae*, *Saccharomyces paradoxus* e *Schizosaccharomyces pombe* realizado pelo algoritmo Clustal através da interface gráfica ClustalX (versão 1.81).

```
C. albicans      -------TTTGACCTCAAATCAGGTAGGACTACCCGCTGAACTTAAGCATATCAATAAGC 60
C. tropicalis    ------CTTTGACCTCAAATCAGGTAGGACTACCCGCTGAACTTAAGCATATCAATAAGC
S. mikatae       --------TTGACCTCAAATCAGGTAGGAGTACCCGCTGAACTTAAGCATATCAATAAGC
S. paradoxus     -------TTTGACCTCAAATCAGGTAGGAGTACCCGCTGAACTTAAGCATATCAATAAGC
S. cerevisiae    ------GTTTGACCTCAAATCAGGTAGGAGTACCCGCTGAACTTAAGCATATCAATAAGC
S. bayanus       -------TTTGACCTCAAATCAGGTAGGAGTACCCGCTGAACTTAAGCATATCAATAAGC
C. glabrata      ------GTTTGACCTCAAATCAGGTAGGGTTACCCGCTGAACTTAAGCATATCAATAAGC
C. neoformans    TTTTTNGTTTGACCTCAAATCAGGTAGGGCTACCCGCTGAACTTAAGCATATCAATAAGC
S. pombe         ------ATTTGACCTCAAATCAGGTAGGACTACGCGCTGAACTTAAGCATATCAATAAGC
A. fumigatus     --------TTGACCTCGGATCAGGTAGGGATACCCGCTGAACTTAAGCATATCAATAAGC
                        ********  **********  *** **************************

C. albicans      GGAGGAAAAGAAACCAACAGGGATTGCCTCAGTAGCGGCGAGTGAAGCGGCAAAAGCTCA 120
C. tropicalis    GGAGGA----AAACCAACAGGNATTGCCTTAGTAGCGGCGAGTGAAGCGGCAAAAGCTCA
S. mikatae       GGAGGAAAAGAAACCAACCGGGATTGCCTTAGTAACGGCGAGTGAAGCGGCAAAAGCTCA
S. paradoxus     GGAGGAAAAGAAACCAACCGGNATTGCCTTAGTAACGGCGAGTGAAGCGGCAAAAGCTCA
S. cerevisiae    GGAGGAAAAGAAACCAACCGGGATTGCCTTAGTAACGGCGAGTGAAGCGGCAAAAGCTCA
S. bayanus       GGAGGAAAAGAAACCAACCGGGATTGCCTTAGTAACGGCGAGTGAAGCGGCAAAAGCTCA
C. glabrata      GGAGGAAAAGAAACCAACTGGGATTGCCTTAGTAACGGCGAGTGAAGCGGCAAAAGCTCA
C. neoformans    GGAGGAAAAGAAACTAACAAGGATTCCCTTAGTAACGGCGAGTGAACCGGGAAGAGCTCA
S. pombe         GCAGGAAAAGAAAATAACCATGATTCCCTCAGTAACGGCGAGTGAAGCGGGAAAAGCTCA
A. fumigatus     GGAGGAAAAGAAACCAACAGGGATTGCCTCAGTAACGGCGAGTGAAGCGGCAAGAGCTCA
                 * ****    ***  ***    *** *** **** *********** *** ** ******

C. albicans      AATTTGAAATCTGGCG------TCTTTGGCGTCCGAGTTGTAATTTGAAGAAGGTATCTT 180
C. tropicalis    AATTTGAAATCTGGCT------CTTTCAGAGTCCGAGTTGTAATTTGAAGAAGGTATCTT
S. mikatae       AATTTGAAATCTGGTA------CCTTTGGTGCCCGAGTTGTAATTTGGAGAGGGCAATTT
S. paradoxus     AATTTGAAATCTGGTA------CCTTCGGTGCCCGAGTTGTAATTTGGAGAGGGCAACTT
S. cerevisiae    AATTTGAAATCTGGTA------CCTTCGGTGCCCGAGTTGTAATTTGGAGAGGGCAACTT
S. bayanus       AATTTGAAATCTGGTA------CCTTTGGTGCCCGAGTTGTAATTTGGAGAGGGCAACTT
C. glabrata      AATTTGAAATCTGGTA------CCTTTGGTGCCCGAGTTGTAATTTGGAGAGTACCACTT
C. neoformans    AATTTGAAATCTGGCG----TCCTCCGGGCGTCCGAGTTCTAATCTACAGAAAC-GTTTT
S. pombe         AATTTGAAATCTGGCAACATTTCTTTTGTTGTCCGAGTTGTAATTTCAAGAAGCTGCTTT
A. fumigatus     AATTTGAAAGCTGGCC------CCTTCGGGGTCCGCGTTGTAATTTGCAGAGGATGCTTC
                 ********* ****              * *** *** **** *  ***       *

C. albicans      TGGGCCCGGCTCTT--GTCTATGTTCCTTGGAACAGGACGTCACAGAGGGTGAGAATCCC 240
C. tropicalis    TGGGTCTGGCTCTT--GTCTATGTTTCTTGGAACAGAACGTCACAGAGGGTGAGAATCCC
S. mikatae       CGAGACCGTTCCTT--GTCTATGTTCCTTGGAACAGGACGTCATAGAGGGTGAGAATCCC
S. paradoxus     TGGGGCCGTTCCTT--GTCTATGTTCCTTGGAACAGGACGTCATAGAGGGTGAGNATCCC
S. cerevisiae    TGGGGCCGTTCCTT--GTCTATGTTCCTTGGAACAGGACGTCATAGAGGGTGAGAATCCC
S. bayanus       TGGGACCGTTCCTT--GTCTATGTTCCTTGGAACAGGACGTCATAGAGGGTGAGAATCCC
C. glabrata      TGGGACTGTACTTT--GCCTATGTTCCTTGGAACAGGACGTCATGGAGGGTGAGAATCCC
C. neoformans    CCGTGCTGGACCGT--GTCTAAGTCCCTTGGAATAGGGTATCAAAGTGGGTGACAATCCC
S. pombe         GAGTG-TAGACGATCGGTCTAAGTTCCTTGGAACAGGACGTCAGAGAGGGTGAGAACCCC
A. fumigatus     GGGTG-CAGCCCCC--GTCTAAGTGCCCTGGAACGGGCCGTCATAGAGGGTGAGAATCCC
                               * *** **  * *****  *    ***  * ******  * ***

C. albicans      GTGCGATGAGATGACCCGGGTCT--GTGTAAAGTTCCTTTGACGAGTCGAGTTGTTTGGG 300
C. tropicalis    GTGCGATGAGATGATCCAGGCCT--ATGTAAAGTTCCTTCGAAGAGTCGAGTTGTTTGGG
S. mikatae       GTGTGGCGAGG-AGTGCGGTTCTCAATGTGAAATGCCTTCGAAGAGTCGAGTTGTTTGGG
S. paradoxus     GTGTGGCGAGG-AGTGCGGTTCT--ATGTAAAGTGCCTTCGAAGAGTCGAGTTGTTTGGG
S. cerevisiae    GTGTGGCGAGG-AGTGCGGTTCT--TTGTAAAGTGCCTTCGAAGAGTCGAGTTGTTTGGG
S. bayanus       GTGTGGCGAGG-AGTGCGGTTCT--ATGTAAAGTGCCTTCGAAGAGTCGAGTTGTTTGGG
C. glabrata      GTGTGGCGAGG-GTGTCAGTTCT--TTGTAAAGGGTGCTCGAAGAGTCGAGTTGTTTGGG
C. neoformans    GTACT-TGACACGATCACCAGTGCTCTGTGATACGTTTTCTACGAGTCGCGTTACTTGGG
S. pombe         GTCTT-TGGT-CGATTGGATATGCCATATAAAGCGCTTTCGAAGAGTCGAGTTGTTTGGG
A. fumigatus     GTCTG-GGACGGGGTGTCTGCGTCCGTGTGAAGCTCCTTCGACGAGTCGAGTTGTTTGGG
                 **     *                    * * *       *  * ****** ***  *****
```

```
C. albicans      AATGCAGCTCTAAGTGGGTGGTAAATTCCATCTAAAGCTAAATATTGGCGAGAGACCGAT 360
C. tropicalis    AATGCAGCTCTAAGTGGGTGGTAAATTCCATCTAAAGCTAAATATTGGCGAGAGACCGAT
S. mikatae       AATGCAGCTCTAAGTGGGTGGTAAATTCCATCTAAAGCTAAATATTGGCGAGAGACCGAT
S. paradoxus     AATGCAGCTCTAAGTGGGTGGTAAATTCCATCTAAAGCTAAATATTGGCGAGAGACCGAT
S. cerevisiae    AATGCAGCTCTAAGTGGGTGGTAAATTCCATCTAAAGCTAAATATTGGCGAGAGACCGAT
S. bayanus       AATGCAGCTCTAAGTGGGTGGTAAATTCCATCTAAAGCTAAATATTGGCGAGAGACCGAT
C. glabrata      AATGCAGCTCTAAGTGGGTGGTAAATTCCATCTAAAGCTAAATACAGGCGAGAGACCGAT
C. neoformans    AGTGTAGCGCAAAATGGGTGGTAAACTCCATCTAAAGCTAAATATTGGTGGAAGACCGAT
S. pombe         AATGCAGCTCTAAATGGGTGGTAAATTTCATCTAAAGCTAAATATTGGCGAGAGACCGAT
A. fumigatus     AATGCAGCTCTAAATGGGTGGTAAATTTCATCTAAAGCTAAATACTGGCCGGAGACCGAT
                 * ** *** * ** *********** * ***************  **    ********

C. albicans      AGCGAACAAGTACAGTGATGGAAAGATGAAAAGAACTTTGAAAAGAGAGTGAAAAAGTAC 420
C. tropicalis    AGCGAACAAGTACAGTGATGGAAAGATGAAAAGAACTTTGAAAAGAGAGTGAAAAAGTAC
S. mikatae       AGCGAACAAGTACAGTGATGGAAAGATGAAAAGAACTTTGAAAAGAGAGTGAAAAAGTAC
S. paradoxus     AGCGAACAAGTACAGTGATGGAAAGATGAAAAGAACTTTGAAAAGAGAGTGAAAAAGTAC
S. cerevisiae    AGCGAACAAGTACAGTGATGGAAAGATGAAAAGAACTTTGAAAAGAGAGTGAAAAAGTAC
S. bayanus       AGCGAACAAGTACAGTGATGGAAAGATGAAAAGAACTTTGAAAAGAGAGTGAAAAAGTAC
C. glabrata      AGCGAACAAGTACAGTGATGGAAAGATGAAAAGAACTTTGAAAAGAGAGTGAAAAAGTAC
C. neoformans    AGCGAACAAGTACCGTGAGGGAAAGATGAAAAGCACTTTGGAAAGAGAGTTAAACAGTAC
S. pombe         AGCGAACAAGTAGAGTGATCGAAAGATGAAAAGAACTTTGAAAAGAGAGTTAAATAGTAC
A. fumigatus     AGCGCACAAGTAGAGTGATCGAAAGATGAAAAGCACTTTGAAAAGAGAGTTAAACAGCAC
                 **** *******  ****  ************* ****** ********* *** ** **

C. albicans      GTGAAATTGTTGAAAGGGAAGGGCTTGAGATCAGACTTGG-TATTTTGCATG--CTGCTC 480
C. tropicalis    GTGAAATTGTTGAAAGGGAAGGGCTTGAGATCAGACTTGG-TATTTTGTATG--TTACTT
S. mikatae       GTGAAATTGTTGAAAGGGAAGGGCATTTGATCAGACATGG-TGTTTTGTGCCCTCTGCTC
S. paradoxus     GTGAAATTGTTGAAAGGGAAGGGCATTTGATCAGACATGG-TGTTTTGTGCCCTCTGCTC
S. cerevisiae    GTGAAATTGTTGAAAGGGAAGGGCATTTGATCAGACATGG-TGTTTTGTGCCCTCTGCTC
S. bayanus       GTGAAATTGTTGAAAGGGAAGGGCATTTGATCAGACATGG-TGTTTTGCGCCCTCTGCTC
C. glabrata      GTGAAATTGTTGAAAGGGAAGGGCATTTGATCAGACATGG-TGTTTTGCGCCCCTTGCCT
C. neoformans    GTGAAATTGTTGAAAGGGAAACGATTGAAGTCAGTCGTGTCTAT-TGGGTTCAGCCAGNT
S. pombe         GTGAAATTGCTGAAAGGGAAGCATTGGAAATCAGTCTTACCTGGGTGAGATCAGTAGTCT
A. fumigatus     GTGAAATTGTTGAAAGGGAAGCGTTTGCGACCAGACTCGCCCGCGGGGTTCAGCCGGCAT
                 ********* **********          *** *

C. albicans      TCT-CGGGG---GCGGCCGCTGCGGTT-TACCGGGCCAGCATCGGTTTGGAGCGGCAGGA 540
C. tropicalis    CTT-CGGGG---GTGGCCTCTACAGTT-TATCGGGCCAGCATCAGTTTGG-GCGGTAGGA
S. mikatae       CTT-GTGGGTAGGGGAATCTCGCATTT-CACTGGGCCAGCATCAGTTTTG-GTGGCAGGA
S. paradoxus     CTT-GTGGGTAGGGGAATCTCGCATTT-CACTGGGCCAGCATCAGTTTTG-GTGGCAGGA
S. cerevisiae    CTT-GTGGGTAGGGGAATCTCGCATTT-CACTGGGCCAGCATCAGTTTTG-GTGGCAGGA
S. bayanus       CTT-GTGGGTGGGGGAATCTCGCAGTT-CACTGGGCCAGCATCAGTTTTG-GTGGCAGGA
C. glabrata      CTC-GTGGGCTTGGGACTCTCGCAGCT-CACTGGGCCAGCATCGGTTTTG-GCGGCCGGA
C. neoformans    CT--GCTGGTGTATTCCCTTTAGAC-------GGGTCAACATCAGTTCTGATCGGTGGAT
S. pombe         CTTCGCGAGACTATGCACTCTGAACCTGTGGTAGGTCAGCATCAGTTTTCGGGGGCGGAA
A. fumigatus     TCGTGCCGGTGTACTTCCCCGTGGGC------GGGCCAGCGTCGGTTTGGGCGGCCGGTC
                       *                       ** ** * ** ***      *   *

C. albicans      TAATGGCGGAGGAATGTGGCACG-----------GCTTCTG--CTGTGTGTTATAGCCT- 600
C. tropicalis    GAATTGCGTTGGAATGTGGCACG-----------NCNTNGG--TTGTGTGTTATAGCCT-
S. mikatae       TAAATCCGTAGGAATGTAACTT------------GCTTCGG---GAAGTATTATAGCCT-
S. paradoxus     TAAATCCGTAGGAATGTAACTT------------GCTTCGG---GAAGTATTATAGCCT-
S. cerevisiae    TAAATCCATAGGAATGTAGCTT------------GCCTCGG---TAAGTATTATAGCCT-
S. bayanus       TAAATCCGTAGGAATGTAACTT------------GCTTCGG---GAAGTATTATAGCCT-
C. glabrata      AAAAACCTAGGGAATGTGGCTCTGC---------GCCTCGGTGTAGAGTGTTATAGCCC-
C. neoformans    AAGGGCTGGAGGAATGTGGCACTCTTCGGGGTGTGTTATAGCCTCCTGTCGCATACACTG
S. pombe         AAAGAATAAGGGAAGGTGG--CTTTCCGGGTTCTGCCTGGGGA--GTGTTTATAGCCCTT
A. fumigatus     AAAGGCCCTCGGAATGTATCAC------------CTCTCGG---GGTGTCTTATAGCCG-
                  *        **** **                        *      **       *

C. albicans      -CTGAC-------GATACTG---CCAGCCTAGACCGAGGACTGCGGTTTTTTT-ACCTAG 660
C. tropicalis    -TCGTC-------GATACTNG--CCAGCCTAGACTGAGGACTGCGGTTTAT---ACCTAG
S. mikatae       -GCGGG-------AATACTG---CCAGCTGGGACTGAGGACTGCGACGTAAGTCA---AG
S. paradoxus     -GCGGG-------AATACTG---CCAGCTGGGACTGAGGACTGCGACGTAAGTCA---AG
S. cerevisiae    -GTGGG-------AATACTG---CCAGCTGGGACTGAGGACTGCGACGTAAGTCA---AG
S. bayanus       -ATGGG-------AATACTG---CCANCTGGGACTGAGGACTGCGACGTAAGTCA---AG
C. glabrata      -TGGGG-------AATACGG---CCAGTCGGGACCGAGGACTGCGATACTTGTTATCTAG
C. neoformans    GTTGGGACTGAGGAATGCAGCTCGCCTTTATGGCCGGGGTTCGCCCACGTTCGAGCTTAG
S. pombe         GTTGT--------AATACGTC---CACTGGGGACTGAGGACTG--CGGCTTCGTGCCAAG
A. fumigatus     AGGGTGC------AATGCGG---CCTGCCTGGACCGAGGAACGCGCTTCGGC-----TCG
                    *          ** *      *      * * * **   *               *
```

```
C. albicans      GATGTTGGCATAATGATCTTAAGTCGCCCGTCTTGAAACACGGACCAAGGAGTCTAACGT 720
C. tropicalis    GATGTTGGCATAATGATCTTAAGTCGCCCGTCTTGAAACACGGACCAAGGAGTCTAACGT
S. mikatae       GATGCTGGCATAATGGTTATATGCCGCCCGTCTTGAAACACGGACCAAGGAGTCTAACGT
S. paradoxus     GATGCTGGCATAATGGTTATATGCCGCCCGTCTTGAAACACGGACCAAGGAGTCTAACGT
S. cerevisiae    GATGCTGGCATAATGGTTATATGCCGCCCGTCTTGAAACACGGACCAAGGAGTCTAACGT
S. bayanus       GATGCTGGCATAATGGTTATATGCCGCCCGTCTTGAAACACGGACCAAGGAGTCTAACGT
C. glabrata      GATGCTGGCATAATGGTTATATGCCGCCCGTCTTGAAACACGGACCAAGGAGTCTAACGT
C. neoformans    GATGTTGACAAAATGGCTTTAAACGACCCGTCTTGAAACACGGACCAAGGAGTCTAACAT
S. pombe         GATGCTGACATAATGGTTTTCAATGGCCCGTCTTGAAACACGGACCAAGGAGTCTAGCAT
A. fumigatus     GACGCTGGCGTAATGGTCGTAAATGACCCGTCTTGAAACACGGACCAAGGAGTCTAACAT
                 ** * ** *  ****    *      ****************************** * *

C. albicans      CTATGCGAGTGTTTGGGTGTA--AAACCCGTACGCGTAATGAAAGTGAACGAAGGTGGGG 780
C. tropicalis    CTATGCGAGTGTTTGGGTGTA--AAACCCGTACGCGTAATGAAAGTGAACGTAGGTGGGG
S. mikatae       CTATGCGAGTGTTTGGGTGTA--AAACCCATACGCGTAATGAAAGTGAACGTAGGTTGGG
S. paradoxus     CTATGCGAGTGTTTGGGTGTA--AAACCCATACGCGTAATGAAAGTGAACGTAGGTTGGG
S. cerevisiae    CTATGCGAGTGTTTGGGTGTA--AAACCCATACGCGTAATGAAAGTGAACGTAGGTTGGG
S. bayanus       CTATGCAAGTGTTTGGGTGTA--AAACCCATACGCGTAATGAAAGTGAACGTAGGTTGGG
C. glabrata      CTATGCGAGTGTTTGGGTGTT--AAACCCGTACGCGTAATGAAAGTGAACGTAGGTTGGG
C. neoformans    ATCTGCGAGTGTTTGAGTGTC--AAACTCGAGCGCGNAATGAAAGTGAATGTAGGAGGGA
S. pombe         CTATGCGAGTGTTTGGGTGATGAAAACCCATCCGCGAAATGAAAGTGAATGCAGGTGGGA
A. fumigatus     CTACGCGAGTGTTCGGGTGTC--AAACCCGTACGCGCAGTGAAAGCGAACGGAGGTGGGA
                  *  ** ****** * ***    **** *   **** * ****** *** * ***  **

C. albicans      GCCCATTA------GGGTGCACCATCGACCGATCCTG-ATGTGTTCGGATGGA----TTT 840
C. tropicalis    GCCCGTAT------GGGTGCACCATCGACCGATCCTG-ATGTCTTCGGATGGA----TTT
S. mikatae       GCCTCGCA----AGAGGTGCACAATCGACCGATCCTG-ATGTCTTCGGATGGA----TTT
S. paradoxus     GCCTCGCA----AGAGGTGCACAATCGACCGATCCTG-ATGTCTTCGGATGGA----TTT
S. cerevisiae    GCCTCGCA----AGAGGTGCACAATCGACCGATCCTG-ATGTCTTCGGATGGA----TTT
S. bayanus       GCCTCGCA----AGAGGTGCACAATCGACCGATCCTG-ATGTCTTCGGATGGA----TTT
C. glabrata      GCCCTCCACCTGGGGGGTGCACAATCGACCGATCCTG-ATGTCTTCGGATGGA----TTT
C. neoformans    TC-----C---GCAAGGAGCACCTTCGACCGATCCGG-ATCTTCTGTGATGGA----TTT
S. pombe         ACGCCCTT---GTGGCGTGCACCATCGACCGACCCGG-AAGTTTGTCAATGGAAGGGTTT
A. fumigatus     GCCCCCTC---GCGGGGCGCACCATCGACCGATCCTGCATGTCTTCGGATGGA----TTT
                  *              * ****  ******** ** * *  *      *****    ***

C. albicans      GAGTAAGAGCATAGCTGTTGGGACCCGAAAGATGGTGAACTATGCCTGAATAGGGTGAAG 900
C. tropicalis    GAGTAAGAGCATAGCTGTTGGGACCCGAAAGATGGTGAACTATGCCTGAATAGGGTGAAG
S. mikatae       GAGTAAGAGCATAGCTGTTGGGACCCGAAAGATGGTGAACTATGCCTGAATAGGGTGAAG
S. paradoxus     GAGTAAGAGCATAGCTGTTGGGACCCGAAAGATGGTGAACTATGCCTGAATAGGGTGAAG
S. cerevisiae    GAGTAAGAGCATAGCTGTTGGGACCCGAAAGATGGTGAACTATGCCTGAATAGGGTGAAG
S. bayanus       GAGTAANAGCATAGCTGTTGGGACCCGAAAGATGGTGAACTATGCCTGAATAGGGTGAAG
C. glabrata      GAGTAAGAGCATAGCTGTTGGGACCCGAAAGATGGTGAACTATGCCTGAATAGGGTGAAG
C. neoformans    GAGTAAGAGCATATATGCTGGGACCCGAAAGATGGTGAACTATGCCTGAATAGGGCGAAG
S. pombe         GAGTAAGAGCATAGCTGTTGGGACCCGAAAGATGGTGAACTATGCCTGAATAGGGTGAAG
A. fumigatus     GAGTACGAGCGTAGCTGTGGGNACCCGAAAGATGGTGAACTATGCCTGAATAGGGTGAAG
                 *****  *** **  **  ** ********************************* ****
```

Anexo 5

Código do programa **jblast2_4.pl**.

Cada linha de código termina com o símbolo ";". Todo o conteúdo de uma linha após o símbolo "#" representa um comentário, isto é: não é interpretado pelo Perl.exe, servindo apenas de informação ao utilizador. A única excepção a esta regra é a primeira linha (#!c:\perl\bin\perl.exe), que identifica o ficheiro como sendo um script de PERL e indica a localização do interpretador.

Este programa pede para se indicar o nome do ficheiro que contém as sequências (em formato FASTA e com a sequência primária numa única linha) a pesquisar. A directoria em que este programa é executado tem que conter uma subdirectoria de nome "sequences". Nesta subdirectoria, encontra-se outra subdirectoria, de nome "database" com os ficheiros resultantes da formatação da base de dados.

```
#!c:\perl\bin\perl.exe
$time_ini = time;

print ("Indica o nome do ficheiro com as seq query:\n");
$seqqueryfile = <STDIN>;
chop ($seqqueryfile);

mkdir ("sequences\\$seqqueryfile") || die ("Unable to create directory sequences\\$seqqueryfile");
mkdir ("sequences\\$seqqueryfile\\queries") || die ("Unable to create directory sequences\\$seqqueryfile\\queries");
mkdir ("sequences\\$seqqueryfile\\blastreports") || die ("Unable to create directory sequences\\$seqqueryfile\\blastreports");
open (SEQFILE, "$seqqueryfile") || die ("Unable to open $seqqueryfile");
open (TOTALQUERIES, ">>sequences\\$seqqueryfile\\queries\\totalqueries.txt") || die ("Unable to open totalqueries.txt");

print TOTALQUERIES ("The file $seqqueryfile: contains the following sequences:\n\n");
$id = 1;
while ($header = <SEQFILE>){
   next if ($header !~ /^>/);
   $skip = tell (SEQFILE);
   print TOTALQUERIES ("\n50 nucleot. query sequences of $header\n");
   $blastthis="";
   until ($blastthis =~ /\n$/) {
      seek (SEQFILE, $skip++, 0);
      read (SEQFILE, $blastthis, 50);
      next if ($blastthis =~ /[^ATGCatgc]/);
      last if (length $blastthis < 50);
      print ("The query sequence is $blastthis\n");
      $queryfile = "sequences\\$seqqueryfile\\queries\\queryseq" . $id . ".txt";
      open (QUERYSEQ, ">$queryfile") || die ("Unable to open $queryfile");
      print QUERYSEQ ("$blastthis");
      close QUERYSEQ;
      print TOTALQUERIES ("$queryfile\t$blastthis\n");
      $reportfile = "sequences\\$seqqueryfile\\blastreports\\report" . $id++ . ".txt";
      system ("blastall -p blastn -d sequences\\database\\database.txt -i $queryfile -o $reportfile");
   }
}
close TOTALQUERIES;
close SEQFILE;
```

```
print ("\nA renomear alguns ficheiros...\n");
opendir (REPORTDIR, "sequences\\$seqqueryfile\\blastreports") || die ("Unable to open directory sequences");
@files = readdir (REPORTDIR);
$biggerlength = 1;
foreach $file (@files) {
   next if (-d "sequences\\$seqqueryfile\\blastreports\\" . $file);
   if (length $file > $biggerlength) {
      $biggerlength = length $file;
   }
}

foreach $file (@files) {
   next if ($file =~ /^\.{1,2}$/);
   next if (-d "sequences\\$seqqueryfile\\blastreports\\" . $file);
   $filename_length = length $file;
   if ($numb_of_zeros = $biggerlength - $filename_length){
      $zeros = 0 x $numb_of_zeros;
      $file_old = $file;
      $file =~ s/(report)/$1$zeros/;
      rename ("sequences\\$seqqueryfile\\blastreports\\$file_old", "sequences\\$seqqueryfile\\blastreports\\$file");
   }
}
closedir (REPORTDIR);

@timelist = localtime (time - $time_ini);
print ("jblast2_4.pl demorou $timelist[2] horas, $timelist[1] minutos e $timelist[0] segundos\n");
```

Anexo 6

Código do programa **parsereport1j.pl**.

Este programa analisa os resultados (ficheiros e directorias) do programa anterior e tem que ser executado na mesma directoria.

```
#!c:\perl\bin\perl.exe
my $time_ini = time;

use strict;
use Bio::SearchIO;

opendir (REPORTS, "sequences\\blastreports") || die ("Unable to open directory blastreports");
my @files = readdir (REPORTS);
closedir (REPORTS);

my @total;
foreach my $file (sort @files) {
   next if ($file =~ /^\.{1,2}$/);
   next if (-d "sequences\\blastreports\\" . $file);
   open (FILE, "sequences\\blastreports\\$file");
   my $in = new Bio::SearchIO(-format => 'blast',
                              -file   => "sequences\\blastreports\\$file");
   while( my $result = $in->next_result ) {
      my $string = "";
      while( my $hit = $result->next_hit ) {
         my $scorebits = $hit->raw_score;
         $string = $string . $scorebits;
      }
      $string = $string . $file;
      push (@total, $string);
   }
}
@total = sort @total;

open (PARSERESULTS, ">sequences\\parseresults.txt") || die ("Unable to open parseresults.txt");
print PARSERESULTS ("*PARSE REPORT*\n\nQueries ordenadas segundo os hit scores:\n");
my $count = 1;
foreach my $string (@total){
   print PARSERESULTS ("$count:\t$string\n");
   $count++;
}
close PARSERESULTS;

my @timelist = localtime (time - $time_ini);
print ("parsereport1j.pl demorou $timelist[2] horas, $timelist[1] minutos e $timelist[0] segundos.\n");
```

Anexo 7

Código do programa **jblast4b.pl**.

Este programa pede para se indicar a directoria onde se encontram os ficheiros que contêm as sequências a pesquisar (em formato FASTA e com a sequência primária numa única linha). Cada ficheiro pode conter mais do que uma sequência. Essa directoria tem que conter uma subdirectoria de nome "database" com os ficheiros resultantes da formatação da base de dados. É igualmente pedido para indicar o tamanho, em nucleótidos, das janelas a pesquisar.

```
#!c:\perl\bin\perl.exe
my $time_ini = time;

use strict;
use Bio::SearchIO;

print ("Indica o path da directoria com os ficheiros das seq query:\n(sem \\ no final)\n");
my $folder = <STDIN>;
chop ($folder);
opendir (FOLDER, "$folder") || die ("Unable to open directory $folder");
my @files = readdir (FOLDER);
closedir (FOLDER);

print ("Indica o tamanho da sonda:\n");
my $janela = <STDIN>;
chop ($janela);

my @files3;
foreach my $seqqueryfile (sort @files) {
   next if ($seqqueryfile =~ /^\.{1,2}$/);
   next if (-d "$folder\\" . $seqqueryfile);
   my $seqqueryfile_ = $seqqueryfile . "_$janela" . "bp";
   my $totalqueries = "$seqqueryfile" . "_totalqueries.txt";
   mkdir ("$folder\\$seqqueryfile_") || die ("Unable to create directory $folder\\$seqqueryfile_");
   mkdir ("$folder\\$seqqueryfile_\\queries") || die ("Unable to create directory $folder\\$seqqueryfile_\\queries");
   mkdir ("$folder\\$seqqueryfile_\\blastreports") || die ("Unable to create directory $folder\\$seqqueryfile_\\blastreports");
   open (SEQFILE, "$folder\\$seqqueryfile") || die ("Unable to open $seqqueryfile");
   open (TOTALQUERIES, ">>$folder\\$seqqueryfile_\\queries\\$totalqueries") || die ("Unable to open $totalqueries");
   print TOTALQUERIES ("The file $seqqueryfile: contains the following sequences:\n\n");
   my $id = 1;
   while (my $header = <SEQFILE>){
      next if ($header !~ /^>/);
      my $skip = tell (SEQFILE);
      print TOTALQUERIES ("\n$janela nucleot. query sequences of $header\n");
      print ("\n\tQuery sequences of $header\n");
      my $blastthis="";
      until ($blastthis =~ /\n$/) {
         seek (SEQFILE, $skip++, 0);
         read (SEQFILE, $blastthis, $janela);
         next if ($blastthis =~ /[^ATGCatgc]/);
         last if (length $blastthis < $janela);
         print ("The query sequence is $blastthis\n");
         my $queryfile = "$folder\\$seqqueryfile_\\queries\\queryseq" . $id . ".txt";
```

```
        open (QUERYSEQ, ">$queryfile") || die ("Unable to open $queryfile");
        print QUERYSEQ ("$header$blastthis");
        close QUERYSEQ;
        my $query_tm = &tm ($blastthis);
        print TOTALQUERIES ("queryseq$id.txt\t$blastthis\tTm: $query_tm ºC\n");
        my $reportfile = "$folder\\$seqqueryfile_\\blastreports\\report" . $id++ . ".txt";
        system ("blastall -p blastn -d $folder\\database\\database.txt -i $queryfile -o $reportfile -W 7 -F F");
    }
  }
  close TOTALQUERIES;
  close SEQFILE;

  print ("\nA renomear ficheiros...\n");
  opendir (REPORTDIR, "$folder\\$seqqueryfile_\\blastreports") || die ("Unable to open directory
$folder\\$seqqueryfile_\\blastreports");
  my @files2 = readdir (REPORTDIR);
  my $biggerlength = 1;
  foreach my $file (@files2) {
    next if (-d "$folder\\$seqqueryfile_\\blastreports\\" . $file);
    if (length $file > $biggerlength) {
      $biggerlength = length $file;
    }
  }

  foreach my $file (@files2) {
    next if ($file =~ /^\.{1,2}$/);
    next if (-d "$folder\\$seqqueryfile_\\blastreports\\" . $file);
    my $filename_length = length $file;
    if (my $numb_of_zeros = $biggerlength - $filename_length){
      my $zeros = 0 x $numb_of_zeros;
      my $file_old = $file;
      $file =~ s/(report)/$1$zeros/;
      rename                                                    ("$folder\\$seqqueryfile_\\blastreports\\$file_old",
"$folder\\$seqqueryfile_\\blastreports\\$file");
    }
  }
  closedir (REPORTDIR);

  #PARSING dos reports:

  print ("A fazer parse aos reports de $seqqueryfile...\n");
  mkdir ("$folder\\$seqqueryfile_\\blastreports\\parsereports") || die ("Unable to create directory
$folder\\$seqqueryfile_\\blastreports\\parsereport2d");
  opendir (REPORTS, "$folder\\$seqqueryfile_\\blastreports") || die ("Unable to open directory
$folder\\$seqqueryfile_\\blastreports");

  my @reports = readdir (REPORTS);
  closedir (REPORTS);
  my @total;
  my %m_count;
  foreach my $file (sort @reports) {
    my $outfile = $file;
    $outfile =~ s/([^\.]+)(\.?[^\.]{0,3})/$1_tms$2/;
    next if ($file =~ /^\.{1,2}$/);
    next if (-d "$folder\\$seqqueryfile_\\blastreports\\" . $file);
    open (INFILE, "$folder\\$seqqueryfile_\\blastreports\\$file") || die ("Unable to open $file");
    open (OUTFILE, ">$folder\\$seqqueryfile_\\blastreports\\parsereports\\$outfile") || die ("Unable to open
$outfile");
```

```
      my $in = new Bio::SearchIO(-format => 'blast',
                          -file   => "$folder\\$seqqueryfile_\\blastreports\\$file");
      while( my $result = $in->next_result ) {
         my $match_count_higher = 0;
         while( my $hit = $result->next_hit ) {
            next if ($result->query_name eq $hit->name);
            my $name = $hit->name;
            my $description = $hit->description;
            my $header2 = "\n$name $description";
            push (@total, $header2);
            while( my $hsp = $hit->next_hsp ) {
               my $querystring = $hsp->query_string;
               my $hitstring = $hsp->hit_string;
               my $match_count = 0;
               if ($querystring eq $hitstring) {
                  $match_count = length $hitstring;
               }else{
                  my @hit = split (//, $hitstring);
                  my @query = split (//, $querystring);
                  my $count = 0;
                  foreach my $nucl (@query) {
                     if (@hit[$count] eq $nucl) {
                        $match_count++;
                     }
                     $count++;
                  }
               }
               my $string = "$hitstring\t- matches: $match_count";
               if ($match_count > $match_count_higher) {
                  $match_count_higher = $match_count;
               }
               push (@total, $string);
            }
         }
         if ($match_count_higher < 10) {
            $match_count_higher =~ s/(\d)/0$1/;
         }
         $m_count{$match_count_higher} = $m_count{$match_count_higher} . "\n<a href = blastreports\\$file>$file</a>  -  <a href = blastreports\\parsereports\\$outfile>Matches de todos os hits inespecíficos</a><br>"
      }
      foreach my $sentence (@total){
         print OUTFILE ("$sentence\n");
      }
      @total = "";
      close OUTFILE;
      close INFILE;
   }

   my $parse_filename = "$seqqueryfile" . "_jblast4b.html";
   open (MCOUNT, ">$folder\\$seqqueryfile_\\$parse_filename") || die ("Unable to open $folder\\$seqqueryfile_\\$parse_filename");
   print MCOUNT ("<HTML>\n<HEAD>\n<TITLE> Resultados do parsing dos BLASTreports de $seqqueryfile </TITLE>\n</HEAD>\n\n<BODY>\n<b><a href=..\\$seqqueryfile>$seqqueryfile</a><br><br>\n<a href=queries\\$totalqueries>Todas as seq query</a><br><br>\nReports cujos hits inespecíficos possuem todos um nº de matches <= que:</b><br>\n");
   my %tm;
```

```
   foreach my $subscript (sort keys(%m_count)) {
      print MCOUNT ("\n<br>\n<b>$subscript:</b><br> $m_count{$subscript}\n");
   }
   print MCOUNT ("</BODY>\n</HTML>");
   close MCOUNT;
   push (@files3, $seqqueryfile);

}

my $folder_new;
if ($folder =~ /\:|\\/) {
   my @words = split (/\\/, $folder);
   $folder_new = pop (@words);
}else{
   $folder_new = $folder;
}

my $summary = "summary_" . "$folder_new" . "_$janela" . ".html";
open (FILES3, ">$folder\\$summary") || die ("Unable to open $folder\\$summary");
print FILES3 ("<HTML>\n<HEAD>\n<TITLE>jblast4b.pl em $folder</TITLE>\n</HEAD>\n\n<BODY>\n<b>Resumo da execução de jblast4b.pl na directoria $folder com janelas de $janela bp:</b><br><br>\n");
foreach my $file3 (@files3) {
   my $file3_ = $file3 . "_$janela" . "bp";
   my $parsefile3 = "$file3" . "_jblast4b.html";
   print FILES3 ("<a href=$file3_\\$parsefile3>Resultados de $file3</a><br><br>\n");
}
print FILES3 ("</BODY>\n</HTML>");
close FILES3;
#Subrotina para calcular Tm:
sub tm {
   my $dna_seq = $_[0];
   $_ = $dna_seq;
   my $number_of_a = tr/aA/aA/;
   my $number_of_t = tr/tT/tT/;
   my $number_of_g = tr/gG/gG/;
   my $number_of_c = tr/cC/cC/;
   my $tm = ($number_of_a * 2) + ($number_of_t * 2) + ($number_of_g * 4) + ($number_of_c * 4);
}

my @timelist = localtime (time - $time_ini);
print ("jblast4b.pl demorou $timelist[2] horas, $timelist[1] minutos e $timelist[0] segundos\n");
```

Anexo 8

Código do programa **jblast3d.pl**.
Os requisitos deste programa são exactamente iguais aos do anterior (jblast4b.pl).

```
#!c:\perl\bin\perl.exe
my $time_ini = time;

use strict;
use Bio::SearchIO;

print ("Indica o path da directoria com os ficheiros das seq query:\n(sem \\ no final)\n");
my $folder = <STDIN>;
chop ($folder);
opendir (FOLDER, "$folder") || die ("Unable to open directory $folder");
my @files = readdir (FOLDER);
closedir (FOLDER);

print ("Indica o tamanho da sonda:\n");
my $janela = <STDIN>;
chop ($janela);

my @files3;
foreach my $seqqueryfile (sort @files) {
   next if ($seqqueryfile =~ /^\.{1,2}$/);
   next if (-d "$folder\\" . $seqqueryfile);
   my $seqqueryfile_ = $seqqueryfile . "_$janela" . "bp";
   my $totalqueries = "$seqqueryfile" . "_totalqueries.txt";
   mkdir ("$folder\\$seqqueryfile_") || die ("Unable to create directory $folder\\$seqqueryfile_");
   mkdir ("$folder\\$seqqueryfile_\\queries") || die ("Unable to create directory $folder\\$seqqueryfile_\\queries");
   mkdir ("$folder\\$seqqueryfile_\\blastreports") || die ("Unable to create directory $folder\\$seqqueryfile_\\blastreports");
   open (SEQFILE, "$folder\\$seqqueryfile") || die ("Unable to open $seqqueryfile");
   open (TOTALQUERIES, ">>$folder\\$seqqueryfile_\\queries\\$totalqueries") || die ("Unable to open $totalqueries");
   print TOTALQUERIES ("The file $seqqueryfile: contains the following sequences:\n\n");
   my $id = 1;
   while (my $header = <SEQFILE>){
      next if ($header !~ />/);
      my $skip = tell (SEQFILE);
      print TOTALQUERIES ("\n$janela nucleot. query sequences of $header\n");
      print ("\n\tQuery sequences of $header\n");
      my $blastthis="";
      until ($blastthis =~ /\n$/) {
         seek (SEQFILE, $skip++, 0);
         read (SEQFILE, $blastthis, $janela);
         next if ($blastthis =~ /[^ATGCatgc]/);
         last if (length $blastthis < $janela);
         print ("The query sequence is $blastthis\n");
         my $queryfile = "$folder\\$seqqueryfile_\\queries\\queryseq" . $id . ".txt";
         open (QUERYSEQ, ">$queryfile") || die ("Unable to open $queryfile");
         print QUERYSEQ ("$header$blastthis");
         close QUERYSEQ;
         my $query_tm = &tm ($blastthis);
         print TOTALQUERIES ("queryseq$id.txt\t$blastthis\tTm: $query_tm ºC\n");
         my $reportfile = "$folder\\$seqqueryfile_\\blastreports\\report" . $id++ . ".txt";
```

```
            system ("blastall -p blastn -d $folder\\database\\database.txt -i $queryfile -o $reportfile -W 8 -F F");
        }
    }
    close TOTALQUERIES;
    close SEQFILE;

    print ("\nA renomear ficheiros...\n");
    opendir (REPORTDIR, "$folder\\$seqqueryfile_\\blastreports") || die ("Unable to open directory $folder\\$seqqueryfile_\\blastreports");
    my @files2 = readdir (REPORTDIR);
    my $biggerlength = 1;
    foreach my $file (@files2) {
        next if (-d "$folder\\$seqqueryfile_\\blastreports\\" . $file);
        if (length $file > $biggerlength) {
            $biggerlength = length $file;
        }
    }

    foreach my $file (@files2) {
        next if ($file =~ /^\.{1,2}$/);
        next if (-d "$folder\\$seqqueryfile_\\blastreports\\" . $file);
        my $filename_length = length $file;
        if (my $numb_of_zeros = $biggerlength - $filename_length){
            my $zeros = 0 x $numb_of_zeros;
            my $file_old = $file;
            $file =~ s/(report)/$1$zeros/;
            rename ("$folder\\$seqqueryfile_\\blastreports\\$file_old", "$folder\\$seqqueryfile_\\blastreports\\$file");
        }
    }
    closedir (REPORTDIR);

    #PARSING dos reports:

    print ("A fazer parse aos reports de $seqqueryfile...\n");
    mkdir ("$folder\\$seqqueryfile_\\blastreports\\parsereports") || die ("Unable to create directory $folder\\$seqqueryfile_\\blastreports\\parsereport2d");
    opendir (REPORTS, "$folder\\$seqqueryfile_\\blastreports") || die ("Unable to open directory $folder\\$seqqueryfile_\\blastreports");

    my @reports = readdir (REPORTS);
    closedir (REPORTS);
    my @total;
    my %goodtm;
    foreach my $file (sort @reports) {
        my $outfile = $file;
        $outfile =~ s/([^\.]+)(\.?[^\.]{0,3})/$1_tms$2/;
        next if ($file =~ /^\.{1,2}$/);
        next if (-d "$folder\\$seqqueryfile_\\blastreports\\" . $file);
        open (INFILE, "$folder\\$seqqueryfile_\\blastreports\\$file") || die ("Unable to open $file");
        open (OUTFILE, ">$folder\\$seqqueryfile_\\blastreports\\parsereports\\$outfile") || die ("Unable to open $outfile");
        my $in = new Bio::SearchIO(-format => 'blast',
                                   -file   => "$folder\\$seqqueryfile_\\blastreports\\$file");
        while( my $result = $in->next_result ) {
            my $tm_higher = 0;
            while( my $hit = $result->next_hit ) {
                next if ($result->query_name eq $hit->name);
```

```
            my $name = $hit->name;
            my $description = $hit->description;
            my $header2 = "\n$name $description";
            push (@total, $header2);
            while( my $hsp = $hit->next_hsp ) {
               my $querystring = $hsp->query_string;
               my $hitstring = $hsp->hit_string;
               if ($querystring ne $hitstring) {
                  my @hit = split (//, $hitstring);
                  my @query = split (//, $querystring);
                  my $count = 0;
                  foreach my $nucl (@query) {
                     if (@hit[$count] ne $nucl) {
                        @hit[$count] = "_";
                     }
                     $count++;
                  }
                  $hitstring = join ("", @hit);
               }
               my $tm = &tm ($hitstring);
               my $string = "$hitstring \t - Tm: $tm ºC";
               if ($tm > $tm_higher) {
                  $tm_higher = $tm;
               }
               if ($hsp->hit_string ne $hsp->query_string) {
                  $string = "$string\tmas tem posições diferentes";
               }
               push (@total, $string);
            }
         }
         $goodtm{$tm_higher} = $goodtm{$tm_higher} . "\n<a href = blastreports\\$file>$file</a>  -  <a href = blastreports\\parsereports\\$outfile>Tm's de todos os hits inespecíficos</a><br>"
      }
      foreach my $sentence (@total){
         print OUTFILE ("$sentence\n");
      }
      @total = "";
      close OUTFILE;
      close INFILE;
   }

   my $parse_filename = "$seqqueryfile" . "_jblast3d.html";
   open (GOODTM, ">$folder\\$seqqueryfile_\\$parse_filename") || die ("Unable to open $folder\\$seqqueryfile_\\$parse_filename");
   print GOODTM ("<HTML>\n<HEAD>\n<TITLE> Resultados do parsing dos BLASTreports de $seqqueryfile </TITLE>\n</HEAD>\n\n<BODY>\n<b><a href=..\\$seqqueryfile>$seqqueryfile</a><br><br>\n<a href=queries\\$totalqueries>Todas as seq query</a><br><br>\nReports cujos hits inespecíficos possuem todos Tm <= que:</b><br>\n");
   my %tm;
   foreach my $subscript (sort keys(%goodtm)) {
      print GOODTM ("\n<br>\n<b>$subscript ºC:</b><br> $goodtm{$subscript}\n");
   }
   print GOODTM ("</BODY>\n</HTML>");
   close GOODTM;
   push (@files3, $seqqueryfile);

}
```

```
my $folder_new;
if ($folder =~ /\:|\\/) {
   my @words = split (/\\/, $folder);
   $folder_new = pop (@words);
}else{
   $folder_new = $folder;
}
my $summary = "summary_" . "$folder_new" . "_$janela" . ".html";
open (FILES3, ">$folder\\$summary") || die ("Unable to open $folder\\$summary");
print FILES3 ("<HTML>\n<HEAD>\n<TITLE>Jblast3d.pl em $folder</TITLE>\n</HEAD>\n\n<BODY>\n<b>Resumo da execução de jblast3d.pl na directoria $folder com janelas de $janela bp:</b><br><br>\n");
foreach my $file3 (@files3) {
   my $file3_ = $file3 . "_$janela" . "bp";
   my $parsefile3 = "$file3" . "_jblast3d.html";
   print FILES3 ("<a href=$file3_\\$parsefile3>Resultados de $file3</a><br><br>\n");
}
print FILES3 ("</BODY>\n</HTML>");
close FILES3;
#Subrotina para calcular Tm:
sub tm {
   my $dna_seq = $_[0];
   $_ = $dna_seq;
   my $number_of_a = tr/aA/aA/;
   my $number_of_t = tr/tT/tT/;
   my $number_of_g = tr/gG/gG/;
   my $number_of_c = tr/cC/cC/;
   my $tm = ($number_of_a * 2) + ($number_of_t * 2) + ($number_of_g * 4) + ($number_of_c * 4);
}

my @timelist = localtime (time - $time_ini);
print ("jblast3d.pl demorou $timelist[2] horas, $timelist[1] minutos e $timelist[0] segundos\n");
```

Anexo 9

Código do programa **tm4.pl**.

Este programa pede para ser indicada uma sequência primária de DNA. Após o cálculo da Tm, pelo método de *Nearest-Neighbour*, apresenta o seu valor directamente na linha de comandos.

```
#!c:\perl\bin\perl.exe

print ("*Calculador de Tm*\nIndicar a sequencia:\n");
$dna_seq = <STDIN>;
chop ($dna_seq);

my $tm;
my $oligo_conc = 50 * 1E-9;
my $salt_conc = 0.05;

if (length ($dna_seq) > 7) {
   my $aa_count;
   while ($dna_seq =~ /AA|TT|UU/ig) {
      $aa_count++;
      pos($dna_seq) = pos($dna_seq) - 1;
   }

   my $gg_count;
   while ($dna_seq =~ /GG|CC/ig) {
      $gg_count++;
      pos($dna_seq) = pos($dna_seq) - 1;
   }

   my @matches = $dna_seq =~ /AT/ig;
   my $at_count = @matches;
   @matches = $dna_seq =~ /AU/ig;
   $at_count += @matches;

   @matches = $dna_seq =~ /TA/ig;
   my $ta_count = @matches;
   @matches = $dna_seq =~ /UA/ig;
   $ta_count += @matches;

   @matches = $dna_seq =~ /CA/ig;
   my $ca_count = @matches;
   @matches = $dna_seq =~ /TG/ig;
   $ca_count += @matches;
   @matches = $dna_seq =~ /UG/ig;
   $ca_count += @matches;

   @matches = $dna_seq =~ /GT/ig;
   my $gt_count = @matches;
   @matches = $dna_seq =~ /AC/ig;
   $gt_count += @matches;
   @matches = $dna_seq =~ /GU/ig;
   $gt_count += @matches;

   @matches = $dna_seq =~ /CT/ig;
   my $ct_count = @matches;
   @matches = $dna_seq =~ /AG/ig;
```

```
$ct_count += @matches;
@matches = $dna_seq =~ /CU/ig;
$ct_count += @matches;

@matches = $dna_seq =~ /GA/ig;
my $ga_count = @matches;
@matches = $dna_seq =~ /TC/ig;
$ga_count += @matches;
@matches = $dna_seq =~ /UC/ig;
$ga_count += @matches;

@matches = $dna_seq =~ /CG/ig;
my $cg_count = @matches;

@matches = $dna_seq =~ /GC/ig;
my $gc_count = @matches;

#Cálculo de DeltaG:
my $delta_g;
$delta_g = -5.0;

$delta_g += 1.2 * $aa_count;
$delta_g += 0.9 * $at_count;
$delta_g += 0.9 * $ta_count;
$delta_g += 1.7 * $ca_count;
$delta_g += 1.5 * $gt_count;
$delta_g += 1.5 * $ct_count;
$delta_g += 1.5 * $ga_count;
$delta_g += 2.8 * $cg_count;
$delta_g += 2.3 * $gc_count;
$delta_g += 2.1 * $gg_count;
print ("deltaG: $delta_g\tKcal/mol\n");

#Cálculo de DeltaH:
my $delta_h;
$delta_h += 8.0 * $aa_count;
$delta_h += 5.6 * $at_count;
$delta_h += 6.6 * $ta_count;
$delta_h += 8.2 * $ca_count;
$delta_h += 9.4 * $gt_count;
$delta_h += 6.6 * $ct_count;
$delta_h += 8.8 * $ga_count;
$delta_h += 11.8 * $cg_count;
$delta_h += 10.5 * $gc_count;
$delta_h += 10.9 * $gg_count;
print ("deltaH: $delta_h\tKcal/mol\n");

#Cálculo de DeltaS:
my $delta_s;
$delta_s += 21.9 * $aa_count;
$delta_s += 15.2 * $at_count;
$delta_s += 18.4 * $ta_count;
$delta_s += 21.0 * $ca_count;
$delta_s += 25.5 * $gt_count;
$delta_s += 16.4 * $ct_count;
$delta_s += 23.5 * $ga_count;
$delta_s += 29.0 * $cg_count;
$delta_s += 26.4 * $gc_count;
```

```
    $delta_s += 28.4 * $gg_count;
    print ("deltaS: $delta_s\tcal/(K*mol)\n");

    #Cálculo de NeighborTm
    my $k = 1/$oligo_conc;
    my $r = 1.987;  #cal/(mole*K);
    my $rlnk = $r * log ($k);
    print ("RlnK: $rlnk\n");

    $tm = 1000 * (($delta_h - 3.4) / ($delta_s + $rlnk));
    $tm += -272.9;  #Graus Kelvin para graus Centígrado
    $tm += 7.21 * log ($salt_conc);
    $tm = int (($tm) + 0.5);    #Arredondamento às unidades
}
print ("Tm: $tm graus.\n");
```

Anexo 10

Código do programa **jblast3d2.pl**.

Os requisitos deste programa são semelhantes aos do jblast4b.pl, mas os argumentos (caminho para a directoria com os ficheiros FASTA e tamanho das janelas) são enviados por meio de um simples formulário HTML, pelo método POST. Este formulário envia o caminho da directoria sob o nome "folder" e o tamanho das janelas sob o nome "janela". A execução deste programa pode ser acompanhada através das informações que são enviadas, em tempo real, para a janela do browser. No final do processamento é apresentado uma hiperligação para o ficheiro HTML resumo.

```
#!c:\perl\bin\perl.exe
my $time_ini = time;

use strict;
use Bio::SearchIO;

print "Content-type: text/html\n\n";
print ("<HTML>\n<HEAD>\n<TITLE>J Probe Design Working...</TITLE>\n</HEAD>\n<BODY>\n");
print ("<center>\n<h1>J Probe Design</h1><p></center>");

if ($ENV{'REQUEST_METHOD'} ne 'POST') {
   print "<p align=center><blink>Invalid input method used<blink><p>\n";
   print "Please use POST method instead of $ENV{'REQUEST_METHOD'}.\n";
   exit 0;
}
my $bytes = $ENV{'CONTENT_LENGTH'};

my $query;
read (STDIN, $query, $bytes);
my @variables = split(/&/, $query);
my $a_variable;
my $var_name;
my $value;
my %form;
foreach $a_variable (@variables) {
   ($var_name, $value) = split (/=/, $a_variable);
   $value =~ tr/\+/ /;
   $form{$var_name} = $value;
}
my $folder = $form{'folder'};
my $janela = $form{'janela'};
$folder =~ s/\%3A/\:/g;
$folder =~ s/\%5C/\\/g;

opendir (FOLDER, "$folder") || die ("Unable to open directory $folder");
my @files = readdir (FOLDER);
closedir (FOLDER);

print ("A directoria contém:<br>\n");
foreach my $file_in_folder (sort @files) {
   next if ($file_in_folder =~ /^\.{1,2}$/);
   next if (-d "$folder\\" . $file_in_folder);
   print ("$file_in_folder<br>\n");
}
```

```
my @files3;
foreach my $seqqueryfile (sort @files) {
   next if ($seqqueryfile =~ /^\.{1,2}$/);
   next if (-d "$folder\\" . $seqqueryfile);
   my $seqqueryfile_ = $seqqueryfile . "_$janela" . "bp";
   my $totalqueries = "$seqqueryfile" . "_totalqueries.txt";
   mkdir ("$folder\\$seqqueryfile_") || die ("Unable to create directory $folder\\$seqqueryfile_");
   mkdir ("$folder\\$seqqueryfile_\\queries") || die ("Unable to create directory
$folder\\$seqqueryfile_\\queries");
   mkdir ("$folder\\$seqqueryfile_\\blastreports") || die ("Unable to create directory
$folder\\$seqqueryfile_\\blastreports");
   open (SEQFILE, "$folder\\$seqqueryfile") || die ("Unable to open $seqqueryfile");
   open (TOTALQUERIES, ">>$folder\\$seqqueryfile_\\queries\\$totalqueries") || die ("Unable to open
$totalqueries");
   print TOTALQUERIES ("The file $seqqueryfile: contains the following sequences:\n\n");
   my $id = 1;
   while (my $header = <SEQFILE>){
      next if ($header !~ /^>/);
      my $skip = tell (SEQFILE);
      print TOTALQUERIES ("\n$janela nucleot. query sequences of $header\n");
      print ("\nA fazer BLAST às sequencias query de $header<p>\n");
      my $blastthis="";
      until ($blastthis =~ /\n$/) {
         seek (SEQFILE, $skip++, 0);
         read (SEQFILE, $blastthis, $janela);
         next if ($blastthis =~ /[^ATGCatgc]/);
         last if (length $blastthis < $janela);
         print ("The query sequence is $blastthis<br>\n");
         my $queryfile = "$folder\\$seqqueryfile_\\queries\\queryseq" . $id . ".txt";
         open (QUERYSEQ, ">$queryfile") || die ("Unable to open $queryfile");
         print QUERYSEQ ("$header$blastthis");
         close QUERYSEQ;
         my $query_tm = &tm ($blastthis);
         print TOTALQUERIES ("queryseq$id.txt\t$blastthis\tTm: $query_tm ºC\n");
         my $reportfile = "$folder\\$seqqueryfile_\\blastreports\\report" . $id++ . ".txt";
         system ("blastall -p blastn -d $folder\\database\\database.txt -i $queryfile -o $reportfile -W 8 -F F");
      }
   }
   close TOTALQUERIES;
   close SEQFILE;

   print ("\nA renomear ficheiros...<p>\n");
   opendir (REPORTDIR, "$folder\\$seqqueryfile_\\blastreports") || die ("Unable to open directory
$folder\\$seqqueryfile_\\blastreports");
   my @files2 = readdir (REPORTDIR);
   my $biggerlength = 1;
   foreach my $file (@files2) {
      next if (-d "$folder\\$seqqueryfile_\\blastreports\\" . $file);
      if (length $file > $biggerlength) {
         $biggerlength = length $file;
      }
   }

   foreach my $file (@files2) {
      next if ($file =~ /^\.{1,2}$/);
      next if (-d "$folder\\$seqqueryfile_\\blastreports\\" . $file);
      my $filename_length = length $file;
      if (my $numb_of_zeros = $biggerlength - $filename_length){
```

```
        my $zeros = 0 x $numb_of_zeros;
        my $file_old = $file;
        $file =~ s/(report)/$1$zeros/;
        rename ("$folder\\$seqqueryfile_\\blastreports\\$file_old",
"$folder\\$seqqueryfile_\\blastreports\\$file");
      }
   }
   closedir (REPORTDIR);

   #PARSING dos reports:

   print ("A fazer parse aos reports de $seqqueryfile...<p>\n");
   mkdir ("$folder\\$seqqueryfile_\\blastreports\\parsereports") || die ("Unable to create directory
$folder\\$seqqueryfile_\\blastreports\\parsereport2d");
   opendir (REPORTS, "$folder\\$seqqueryfile_\\blastreports") || die ("Unable to open directory
$folder\\$seqqueryfile_\\blastreports");

   my @reports = readdir (REPORTS);
   closedir (REPORTS);
   my @total;
   my %goodtm;
   foreach my $file (sort @reports) {
      my $outfile = $file;
      $outfile =~ s/([^\.]+)(\.?[^\.]{0,3})/$1_tms$2/;
      next if ($file =~ /^\.{1,2}$/);
      next if (-d "$folder\\$seqqueryfile_\\blastreports\\" . $file);
      open (INFILE, "$folder\\$seqqueryfile_\\blastreports\\$file") || die ("Unable to open $file");
      open (OUTFILE, ">$folder\\$seqqueryfile_\\blastreports\\parsereports\\$outfile") || die ("Unable to open
$outfile");
      my $in = new Bio::SearchIO(-format => 'blast',
                        -file   => "$folder\\$seqqueryfile_\\blastreports\\$file");
      while( my $result = $in->next_result ) {
         my $tm_higher = 0;
         while( my $hit = $result->next_hit ) {
            next if ($result->query_name eq $hit->name);
            my $name = $hit->name;
            my $description = $hit->description;
            my $header2 = "\n$name $description";
            push (@total, $header2);
            while( my $hsp = $hit->next_hsp ) {
               my $querystring = $hsp->query_string;
               my $hitstring = $hsp->hit_string;
               if ($querystring ne $hitstring) {
                  my @hit = split (//, $hitstring);
                  my @query = split (//, $querystring);
                  my $count = 0;
                  foreach my $nucl (@query) {
                     if (@hit[$count] ne $nucl) {
                        @hit[$count] = "_";
                     }
                     $count++;
                  }
                  $hitstring = join ("", @hit);
               }
               my $tm = &tm ($hitstring);
               my $string = "$hitstring \t - Tm: $tm ºC";
               if ($tm > $tm_higher) {
                  $tm_higher = $tm;
```

```
                }
                if ($hsp->hit_string ne $hsp->query_string) {
                   $string = "$string\tmas tem posições diferentes";
                }
                push (@total, $string);
             }
          }
          $goodtm{$tm_higher} = $goodtm{$tm_higher} . "\n<a href =
blastreports\\$file>$file</a>  -  <a href = blastreports\\parsereports\\$outfile>Tm's de
todos os hits inespecíficos</a><br>"
       }
       foreach my $sentence (@total){
          print OUTFILE ("$sentence\n");
       }
       @total = "";
       close OUTFILE;
       close INFILE;
    }

    my $parse_filename = "$seqqueryfile" . "_jblast3d2.html";
    open (GOODTM, ">$folder\\$seqqueryfile_\\$parse_filename") || die ("Unable to open
$folder\\$seqqueryfile_\\$parse_filename");
    print GOODTM ("<HTML>\n<HEAD>\n<TITLE> Resultados do parsing dos BLASTreports de
$seqqueryfile </TITLE>\n</HEAD>\n\n<BODY>\n<b><a
href=..\\$seqqueryfile>$seqqueryfile</a><br><br>\n<a href=queries\\$totalqueries>Todas as seq
query</a><br><br>\nReports cujos hits inespecíficos possuem todos Tm <= que:</b><br>\n");
    my %tm;
    foreach my $subscript (sort keys(%goodtm)) {
       print GOODTM ("\n<br>\n<b>$subscript ºC:</b><br> $goodtm{$subscript}\n");
    }
    print GOODTM ("</BODY>\n</HTML>");
    close GOODTM;
    push (@files3, $seqqueryfile);

}

my $folder_new;
if ($folder =~ /\:|\\/) {
    my @words = split (/\\/, $folder);
    $folder_new = pop (@words);
}else{
    $folder_new = $folder;
}
my $summary = "summary_" . "$folder_new" . "_$janela" . ".html";
open (FILES3, ">$folder\\$summary") || die ("Unable to open $folder\\$summary");
print FILES3 ("<HTML>\n<HEAD>\n<TITLE>Jblast3d2.pl em
$folder</TITLE>\n</HEAD>\n\n<BODY>\n<b>Resumo da execução de jblast3d2.pl na directoria $folder
com janelas de $janela bp:</b><br><br>\n");
foreach my $file3 (@files3) {
    my $file3_ = $file3 . "_$janela" . "bp";
    my $parsefile3 = "$file3" . "_jblast3d2.html";
    print FILES3 ("<a href=$file3_\\$parsefile3>Resultados de $file3</a><br><br>\n");
}
print FILES3 ("</BODY>\n</HTML>");
close FILES3;
#Subrotina para calcular Tm:
sub tm {
    my $dna_seq = $_[0];
```

```
    $_ = $dna_seq;
    my $number_of_a = tr/aA/aA/;
    my $number_of_t = tr/tT/tT/;
    my $number_of_g = tr/gG/gG/;
    my $number_of_c = tr/cC/cC/;
    my $tm = ($number_of_a * 2) + ($number_of_t * 2) + ($number_of_g * 4) + ($number_of_c * 4);
}

my @timelist = localtime (time - $time_ini);
print ("jblast3d2.pl demorou $timelist[2] horas, $timelist[1] minutos e $timelist[0] segundos<p>\n");

print ("<center><h2><a href=$folder\\$summary target=\"_blank\">Resultados</a></h2>\n</center>\n</body>\n</html>");
```

Anexo 11

Código do programa **jsearch9.pl**.

Este programa pede para se indicar o nome do ficheiro que contêm as sequências a pesquisar (em formato FASTA e com a sequência primária numa única linha), o nome do ficheiro com a base de dados, o tamanho das janelas e o tamanho do maior *match* de nucleótidos contíguos a considerar na segunda pesquisa.

```
#!c:\perl\bin\perl.exe
$time_ini = time;

print ("Indica o nome do ficheiro com as seq:\n");
$seqfile = <STDIN>;
chop ($seqfile);
$seqfile2 = $seqfile . "_jsearch9.txt";

print ("\nIndica o nome do ficheiro com a base de dados:\n(Default = database.txt)\n");
$database_file = <STDIN>;
chop ($database_file);

if (length ($database_file) == 0) {
   $database_file = "database.txt"
}

print ("\nIndica o tamanho da janela:\n");
$janela = <STDIN>;
chop ($janela);

print ("\nIndica o tamanho do maior match (de nucl contiguos) a ser considerado:\n");
$maior_match = <STDIN>;
chop ($maior_match);

$range_substcript1 = int ($janela / 2) - 2;
$range_substcript2 = int ($janela / 2) + 2;
$range_substcript3 = $janela - 1;

open (SEQFILE, "$seqfile") || die ("Unable to open $seqfile");
open (TOTALSEARCH, ">$seqfile2") || die ("Unable to open $seqfile2");
open (DATABASE, "$database_file") || die ("Unable to open database.txt");

@database = <DATABASE>;
$database = join("", @database);
$database = uc($database);

print TOTALSEARCH ("jsearch9.pl\n\n");
print TOTALSEARCH ("janelas de $janela nucl. de $seqfile q são específicas e têm um maior match (de nucl contíguos) menor q $maior_match em relação às seq de $database_file:\n");
print ("\nA procurar janelas de $janela nucleotidos em $database_file...\n\n");

while ($header = <SEQFILE>){
   chomp $header;
   next if ($header !~ /^>/);
   $skip = tell (SEQFILE);
   print TOTALSEARCH ("\n$header\n");
   print ("de $header\n");
   $searchthis ="";
   $id = 0;
```

```
   until ($searchthis =~ /\n$/) {
      seek (SEQFILE, $skip++, 0);
      read (SEQFILE, $searchthis, $janela);
      next if ($searchthis =~ /[^ATGCatgc]/);
      last if (length $searchthis < $janela);
      $seq_count = 0;
      while ($database =~ /$searchthis/ig) {
         pos($database) = pos($database) - ($janela - 1);
         $seq_count++;
      }
      $id++;

      if ($seq_count == 1) {
         $searchthis = uc($searchthis);
         $database_ = $database;
         $database_ =~ s/$searchthis//ig;
         $max_match = 0;
         $max_match2 = 0;
         for ($janela2 = $janela - 1 ; $janela2 > 1 ; $janela2--) {
            last if ($janela2 < $max_match);
            $skip2 = 0;
            while ($segment = substr ($searchthis, $skip2++, $janela2)) {
               last if (length ($segment) != $janela2);
               if ($database_ =~ /$segment/i) {
                  $max_match = $janela2;
               }
            }
         }
         if ($max_match <= $maior_match) {
            print TOTALSEARCH ("$searchthis\n");
         }
      }
   }
}
close TOTALSEARCH;
close SEQFILE;

@timelist = localtime (time - $time_ini);
print ("jsearch9.pl demorou $timelist[2] horas, $timelist[1] minutos e $timelist[0] segundos\n");
```

Anexo 12

Código do programa **compmultalign4.pl**.

Este programa trabalha unicamente sobre um qualquer ficheiro de alinhamentos produzidos pelo Clustal. Para além de se indicar o seu nome é, também, necessário indicar o número de sequências que lhe deram origem.

```
#!c:\perl\bin\perl.exe
$time_ini = time;

print ("Indica o nome do ficheiro com o alinhamento:\n");
$seqfile1 = <STDIN>;
chop ($seqfile1);
$seqfile1 =~ /\.aln$/ || die ("Ficheiro n é um ficheiro de alinhamentos de Clustal");
$seqfile2 = $seqfile1;
$seqfile2 =~ s/([^\.]+\.aln)/$1_compmultalign4\.txt/;

print ("\nIndica quantas sequências tem o alinhamento múltiplo:\n");
$numero_seq = <STDIN>;
chop ($numero_seq);

open (FILE, "$seqfile1") || die ("Unable to open $seqfile1");
open (OUTFILE, ">out.txt") || die ("Unable to open out.txt");
open (RESULTS, ">$seqfile2") || die ("Unable to open $seqfile2");

print RESULTS ("Análise do alinhamento múltiplo (CLUSTAL X) do file $seqfile1\n\n");

$trash = <FILE>;
$trash = <FILE>;
$trash = <FILE>;
while ($trash =~ /\n/ ) {
   for ($count = 1 ; $count <= $numero_seq ; $count++) {
      $line = <FILE>;
      chomp $line;
      $header = $line;
      $header =~ s/([\w]+)\s.*/$1/;
      $line =~ s/[\w]+\s*(.*)/$1/;
      $hash{$header} = $hash{$header} . $line;
   }
   $stars = <FILE>;
   $trash = <FILE>;
}
close FILE;

foreach $subscript (sort keys(%hash)) {
   @{$hash{$subscript}} = split (//,$hash{$subscript});
}

foreach my $subscript (sort keys(%hash)) {
   print OUTFILE (">$subscript\n$hash{$subscript}\n\n");
}
close OUTFILE;

foreach $subscript (sort keys %hash) {
   for ($count2 = 0;$count2 < @{$hash{$subscript}} ;$count2++) {
      next if (${$hash{$subscript}}[$count2] =~ /[^AaTtGgCc]/);
      $equal = 0;
```

```
      foreach $subscript2 (sort keys %hash) {
         next if ($subscript eq $subscript2);
         if (${$hash{$subscript}}[$count2] eq ${$hash{$subscript2}}[$count2]) {
            $equal = 1;
         }
      }
      if ($equal == 0) {
         push (@{$hash2{$subscript}}, $count2 + 1);
      }
   }
}

foreach $subscript (sort keys %hash2) {
   $first_done = 1;
   print RESULTS ("\n$subscript:\n");
   $smaller_interspace = 999999999999999999;
   foreach $number (@{$hash2{$subscript}}) {
      print RESULTS ("\n$number:\n");
      $sum = 0;
      $minus_number = $number;
      $plus_number = $number;
      foreach $subscript2 (sort keys %hash2) {
         next if ($subscript eq $subscript2);
         print RESULTS ("$subscript2: ");
         $smaller_distance = 999999999999999999;
         foreach $number2 (@{$hash2{$subscript2}}) {
            $distance = abs ($number - $number2);
            if ($distance < $smaller_distance){
               $nearest_number = $number2;
               $smaller_distance = $distance;
            }
         }
         print RESULTS ("nearest_number: $nearest_number\tsmaller_distance: $smaller_distance\n");
         if ($nearest_number < $minus_number) {
            $minus_number = $nearest_number;
         }
         if ($nearest_number > $plus_number) {
            $plus_number = $nearest_number;
         }
         $interspace = $plus_number - $minus_number + 1;
      }
      print RESULTS ("Intervalo: $minus_number - $plus_number = $interspace nucl.\n");
      if ($interspace < $smaller_interspace) {
         $number_string = "$minus_number - $plus_number";
         $smaller_interspace = $interspace;
      }
   }
   print RESULTS ("\n\nO menor intervalo é de $smaller_interspace nucl, entre as posições
$number_string.\n");
}
close RESULTS;

@timelist = localtime (time - $time_ini);
print ("compmultalign4.pl demorou $timelist[2] horas, $timelist[1] minutos e $timelist[0] segundos\n");
```

Anexo 13

Código do programa **jqblast2.pl**.

Este programa tem que ser executado num computador com acesso à internet. Caso as configurações da *proxy* não correspondam às indicadas (configurações actuais da Universidade de Aveiro), será necessário editar a linha 8. A configuração do ficheiro pedido é a seguinte: um, ou mais, cabeçalhos (iniciados por ">") seguidos das sondas correspondentes, uma por linha.

```
#!c:\perl\bin\perl.exe
$time_ini = time;

use LWP::UserAgent;

$ua = new LWP::UserAgent;
$ua->agent("simple_bot/0.1");
$ua->proxy(http =>'http://proxy.ua.pt:3128');

print ("Indica o nome do ficheiro com as possiveis sondas:\n");
$probefile = <STDIN>;
chop ($probefile);
$ridfile = $probefile . "_rids.txt";

open (PROBES, "<$probefile") || die ("Unable to open $probefile");
open (RIDS, ">$ridfile") || die ("Unable to open $ridfile");

while ($line = <PROBES>){
   chomp ($line);
   next if ($line eq "");
   if ($line =~ /^>/) {
      $header = $line;
      print RIDS ("\n$header\n");
      next;
   }
   print ("The Query sequence is: $line\n");
   print RIDS ("$line\t\t");

$content="CMD=Put&LAYOUT=OneWindow&AUTO_FORMAT=Semiauto&DATABASE=nr&ENTREZ
_QUERY=Fungi+[ORGN]&FILTER=L&PAGE=Nucleotides&PROGRAM=blastn&QUERY=" . $line;
   my $req = new HTTP::Request POST => 'http://www.ncbi.nlm.nih.gov/blast/Blast.cgi';
   $req->content_type('application/x-www-form-urlencoded');
   $req->content($content);
   open (FILE, ">RID_page.html") || die "Error opening RID_page.html: $!";
   my $res = $ua->request($req);
   if ($res->is_success) {
      print FILE $res->content ;
      print "Saving RID page...\n";
   } else {
      print "User agent found an error\n";
   }
   close(FILE);

   #PARSING da página HTML para extrair RID
   print "Parsing RID_page.html...\n";
   open (FILE, "<RID_page.html") || die "Error opening RID_page.html: $!";
   $line2 = 0;
```

```
    $rid ="";
    while ($line2 = <FILE>) {
        if ($line2 =~/^The request ID is/) {
            $line2 =~ /(\d{10}-\d{3,5}-\d{9,12}.BLASTQ\d)/;
            $rid = $&;
        }
    }
    close (FILE);
    print ("o RID e' $rid\n\n");
    print RIDS ("RID: $rid\n");
    sleep (4);
}
close (RIDS);
close (PROBES);

@timelist = localtime (time - $time_ini);
print ("jqblast2.pl demorou $timelist[2] horas, $timelist[1] minutos e $timelist[0] segundos\n");
```

Anexo 14

Código do programa **jqblast2b.pl**.

Os requisitos deste programa em relação à internet e ao *proxy* são iguais ao do anterior (jqblast2.pl). A configuração do ficheiro pedido é semelhante mas, a seguir a cada sonda (na mesma linha), está indicado o identificador único da sua pesquisa (RID) nos servidores do BLAST do NCBI.

```
#!c:\perl\bin\perl.exe
$time_ini = time;

use LWP::UserAgent;

$ua = new LWP::UserAgent;
$ua->agent("simple_bot/0.1");
$ua->proxy(http =>'http://proxy.ua.pt:3128');

print ("Indica o nome do ficheiro com os RID's:\n");
$ridfile = <STDIN>;
chop ($ridfile);
$summary = $ridfile . "_jqblast2b_summary.html";
$folder = $ridfile . "_BLASTS";

open (RIDS, "<$ridfile") || die ("Unable to open $ridfile");
open (SUMMARY, ">$summary") || die ("Unable to open $summary");

mkdir ("$folder") || die ("Unable to create directory $folder");
print SUMMARY ("<html>\n<head>\n<title>Summary</title>\n</head>\n<body>\n");

while ($line = <RIDS>){
   chomp ($line);
   next if ($line eq "");
   if ($line =~ /^>/) {
      $header = $line;
      print SUMMARY ("<br>$header<br>\n");
      next;
   }
   $line =~ s/(\d{10}-\d{3,5}-\d{9,12}.BLASTQ\d)//;
   $rid = $&;
   $line =~ s/\t\tRID: //;
   $probe = $line;
   print ("\nThe RID is: $rid\n");
   $rid_link = $rid;
   $rid_link =~ s/\./_/;
   $rid_link2 = $rid_link . ".html";
   $rid_link3 = $rid_link . "_TaxReport.html";

   #Extrair o Report "Normal":
   $content = "CMD=Get&AUTO_FORMAT=Fullauto&RID=" . "$rid";
   $status = "waiting";
   $loop_iteration1 = 0;
   until ($status eq "ready") {
      last if ($loop_iteration1++ == 5);
      my $req = new HTTP::Request POST => 'http://www.ncbi.nlm.nih.gov/blast/Blast.cgi';
      $req->content_type('application/x-www-form-urlencoded');
      $req->content($content);
      open (REPORT, ">$folder\\$rid_link2") || die "Error opening $folder\\$rid_link2: $!";
```

```
    my $res = $ua->request($req);
    if ($res->is_success) {
        print REPORT $res->content ;
        print "Saving Report page...\n";
    } else {
        print "User agent found an error\n";
    }
    close(REPORT);

    open (REPORT, "<$folder\\$rid_link2") || die "Error opening $folder\\$rid_link2: $!";
    while ($line2 = <REPORT>) {
        if ($line2 =~ /\tStatus=/) {
            if ($line2 =~ /Status=READY/) {
                $status = "ready";
            }
            last;
        }
    }
    close(REPORT);
    sleep (60);
}


#Extrair o Taxonomy report:

$content = "CMD=Get&AUTO_FORMAT=Fullauto&FORMAT_OBJECT=TaxBlast&RID=" . "$rid";
$status2 = "waiting";
$loop_iteration2 = 0;
$count = 0;
until ($status2 eq "ready") {
    last if ($loop_iteration2++ == 5);
    my $req = new HTTP::Request POST => 'http://www.ncbi.nlm.nih.gov/blast/Blast.cgi';
    $req->content_type('application/x-www-form-urlencoded');
    $req->content($content);
    open (TAXREPORT, ">$folder\\$rid_link3") || die "Error opening $folder\\$rid_link3: $!";
    my $res = $ua->request($req);
    if ($res->is_success) {
        print TAXREPORT $res->content ;
        print "Saving TaxReport page...\n";
    } else {
        print "User agent found an error\n";
    }
    close(TAXREPORT);

    open (TAXREPORT, "<$folder\\$rid_link3") || die "Error opening $folder\\$rid_link3: $!";
    while ($line3 = <TAXREPORT>) {
        if ($line3 =~ /\tStatus=/) {
            if ($line3 =~ /Status=READY/) {
                $status2 = "ready";
            }
            last;
        }
    }
    close(TAXREPORT);

    #PARSING do Taxonomy report:
    if ($status2 eq "ready") {
        open (TAXREPORT, "<$folder\\$rid_link3") || die "Error opening $folder\\$rid_link3: $!";
```

```
            $count = 0;
            $especies ="";
            while ($line4 = <TAXREPORT>) {
               if ($line4 =~ /\s{3}30\s{1,2}<a\shref=\#\d+>\d{1,3}\shits?<\/a>/i) {
                  $count++;
                  $line4 =~ />([^<]+)</;
                  $especie = $1;
                  push (@{$hash{$header}{$probe}}, $especie);
               }
            }
            close(TAXREPORT);
         }

         if ($count == 0) {
            $status2 = "waiting";
         }
         sleep (60);
      }

      if ($count == 0) {
         $font_color = "#000000";
         $msg_erro = " ERRO!!!";
         $ss = "s";
      }elsif ($count == 1) {
         $font_color = "#1BE511";
         $msg_erro = "";
         $ss = "";
      }else{
         $font_color = "#FB0915";
         $msg_erro = "";
         $ss = "s";
      }
      print SUMMARY ("<font face=courier color=$font_color>$probe <a
href=\"$folder\\$rid_link2\">Report</a> - <a href=\"$folder\\$rid_link3\">Tax
Report</a>  - &nbspSonda específica para $count espécie$ss.$msg_erro</font><br>\n");
      if ($count == 1) {
         $delete{$header} = 1;
      }
   }

   foreach $header (sort keys %delete) {
      undef %{$hash{$header}};
   }

   $textfile = $ridfile . "_jqblast2b.txt";
   open (TEXTFILE, ">$textfile") || die "Error opening $textfile";

   foreach $header (sort keys %hash) {
      print TEXTFILE ("\n\n$header");
      foreach $probe (sort keys %{$hash{$header}}) {
         print TEXTFILE ("\n$probe");
         foreach $spp (@{$hash{$header}{$probe}}) {
            print TEXTFILE ("$spp", "::");
         }
      }
   }
   close (TEXTFILE);
   print SUMMARY ("</body>\n</html>");
```

```
close (SUMMARY);
close (RIDS);

@timelist = localtime (time - $time_ini);
print ("jqblast2b.pl demorou $timelist[2] horas, $timelist[1] minutos e $timelist[0] segundos\n");
```

Anexo 15

Código do programa **j14b.pl**.

A configuração do ficheiro pedido é semelhante à do ficheiro pedido para o programa jqblast2.pl mas, imediatamente a seguir a cada sonda (na mesma linha), estão indicados os nomes das espécies (separadas por "::") que contêm essa sequência no seu genoma.

```
#!c:\perl\bin\perl.exe
$time_ini = time;

print ("Indica o nome do ficheiro com as especies de cada sonda:\n");
$sppfile = <STDIN>;
chop ($sppfile);
$outfile = $sppfile . "_jac14b.txt";

open (FILE, "<$sppfile") || die "Error opening $sppfile";
open (OUTFILE, ">$outfile") || die "Error opening $outfile";

while ($line = <FILE>) {
   chomp ($line);
   next if ($line eq "");
   if ($line =~ /^>/) {
      $header = $line;
      next;
   }
   $line =~ s/^(\w{15})//;
   $probe = $&;
   $especies = $line;
   @especies = split (::, $especies);
      foreach $especie (@especies) {
         push (@{$hash{$header}{$probe}}, $especie);
      }
}

print OUTFILE ("\nCOMBINAÇÕES DE SONDAS Q IDENTIFICAM ESPÉCIES\n");

foreach $header (sort keys %hash) {
   print ("\n\tde $header\n");
   print OUTFILE ("\n$header\n");
   $id = 0;
   foreach $probe1 (sort keys %{$hash{$header}}) {
      foreach $probe2 (sort keys %{$hash{$header}}) {
         next if ($probe2 eq $probe1);
         foreach $probe3 (sort keys %{$hash{$header}}) {
            next if ($probe3 eq $probe1 || $probe3 eq $probe2);
            foreach $probe4 (sort keys %{$hash{$header}}) {
               next if ($probe4 eq $probe1 || $probe4 eq $probe2 || $probe4 eq $probe3);
               push (@combination, $probe1, $probe2, $probe3, $probe4);
               @combination = sort (@combination);
               $four_probes = join("", @combination);
               undef @combination;
               if ($all_combinations{$four_probes} != 1) {
                  $all_combinations{$four_probes} = 1;
                  $id++;
                  print ("$header - $id\n");
                  undef %hash2;
```

```
                @{$hash2{$probe1}} = @{$hash{$header}{$probe1}};
                @{$hash2{$probe2}} = @{$hash{$header}{$probe2}};
                @{$hash2{$probe3}} = @{$hash{$header}{$probe3}};
                @{$hash2{$probe4}} = @{$hash{$header}{$probe4}};

                $nbr_spp_present_in_4_probes = 0;
                $exit_loop = 0;
                $combination_probes = "";
                foreach $probe (sort keys %hash2) {
                   $combination_probes = $combination_probes . $probe . " ";
                   next if ($exit_loop == 1);
                   $exit_loop = 1;
                   foreach $spp (@{$hash2{$probe}}) {
                      $spp_count = 0;
                      foreach $probe2 (sort keys %hash2) {
                         foreach $spp2 (@{$hash2{$probe2}}) {
                            $spp_count++ if ($spp2 eq $spp);
                         }
                      }
                      if ($spp_count == 4) {
                         $spp_present_in_4_probes = $spp;
                         $nbr_spp_present_in_4_probes++;
                      }
                   }
                }
                chop $combination_probes;
                if ($nbr_spp_present_in_4_probes == 1) {
                   print OUTFILE ("Combinação nº $id é específica para
$spp_present_in_4_probes\n($combination_probes)\n\n");
                }
             }
          }
       }
    }
 }
}
close (OUTFILE);
close (FILE);

@timelist = localtime (time - $time_ini);
print ("j14b.pl demorou $timelist[2] horas, $timelist[1] minutos e $timelist[0] segundos\n");
```

Anexo 16

Código do programa **primers3.pl**.

O ficheiro pedido por este programa tem que possuir a seguinte configuração: um, ou mais, cabeçalhos (iniciados por ">"), cada um seguido de uma, ou mais, sequências (uma por linha). É ainda pedido para indicar o tamanho das janelas a caracterizar como sendo possíveis *primers* (ou sondas).

```
#!c:\perl\bin\perl.exe

print ("\n*primers3.pl*\nCalculador de Tm e Self Anneal\nIndica o nome do ficheiro com as zonas de escolha de primers:\n");
$seqfile = <STDIN>;
chop ($seqfile);

print ("\nIndique o tamanho do primer:\n");
$primer_size = <STDIN>;
chop $primer_size;

$seqfile2 = $seqfile;
$seqfile2 =~ s/([^\.]+)(\.?[^\.]{0,3})/$1_primer3_$primer_size$2/;
open (INFILE, "$seqfile") || die ("Unable to open $seqfile");
open (OUTFILE, ">$seqfile2") || die ("Unable to open $seqfile2");

while ($line = <INFILE>) {
   chomp $line;
   if ($line =~ /^>/) {
      $header = $line;
      print OUTFILE ("\nPrimers em $header:\n");
      next;
   }

   $count = 0;
   while (1) {
      $primer_candidate = substr ($line, $count++, $primer_size);
      last if (length $primer_candidate < $primer_size);
      $tm = &tm ($primer_candidate);
      print OUTFILE ("$primer_candidate  Tm: $tm ºC.\n");
      $tm_higher = 0;
      $string = "";
      for ($janela = 2; $janela <= ($primer_size / 2); $janela++) {
         $count2 = 0;
         while (($count2 + $janela) <= $primer_size) {
            undef @numbers;
            for ($digit = $count2; $digit < ($count2 + $janela); $digit++) {
               push (@numbers, $digit);
            }
            $subseq = substr ($primer_candidate, $count2++, $janela);
            $subseq2 = reverse $subseq;
            $subseq2 =~ tr/AaTtGgCc/TtAaCcGg/;
            $position = 0;

      LOOP:    while ($position < ((length ($primer_candidate)) - (length ($subseq))) && $position != -1) {
                  $position = index($primer_candidate, $subseq2, $position);
                  if ($position != -1) {
                     undef @numbers2;
                     for ($digit2 = $position; $digit2 < ($position + $janela); $digit2++) {
```

```
                push (@numbers2, $digit2);
            }
            $position++;

            foreach $digit3 (@numbers) {
                foreach $digit4 (@numbers2) {
                    if ($digit3 == $digit4 || $digit3 > $digit4) {
                        next LOOP;
                    }
                }
            }

            $numbers = join (", ", @numbers);
            $numbers2 = join (", ", @numbers2);
            $tm2 = &tm ($subseq2);
            if ($tm2 > $tm_higher) {
                $tm_higher = $tm2;
                $string = "$subseq ($numbers)\t$subseq2 ($numbers2)\n";
            }elsif ($tm2 == $tm_higher) {
                $string = $string . "$subseq ($numbers)\t$subseq2 ($numbers2)\n";
            }
        }
    }
}

    }
    print OUTFILE ("Tm máx de self an.: $tm_higher ºC -\nem:\t\tcom:\n$string\n")
  }
}
#Subrotina para calcular Tm:
sub tm {
  my $dna_seq = $_[0];
  $_ = $dna_seq;
  my $number_of_a = tr/aA/aA/;
  my $number_of_t = tr/tT/tT/;
  my $number_of_g = tr/gG/gG/;
  my $number_of_c = tr/cC/cC/;
  my $tm = ($number_of_a * 2) + ($number_of_t * 2) + ($number_of_g * 4) + ($number_of_c * 4);
}
```

Referências

1. McGinnis, M.R., et al., *Mycology*, in *Medical Microbiology*, S. Baron and M.R. McGinnis, Editors. 1996, University of Texas Medical Branch: Galveston.
2. Carlile, M.J. and S.C. Watkinson, *The Fungi*. 1st ed. 1994, London: Academic Press Limited.
3. Deacon, J.W., *Modern Mycology*. 3rd ed. 1997, Oxford: Blackwell Science.
4. Hudson, H.J., *Fungal Biology*. 1st ed. 1986, Cambridge: Cambridge University Press.
5. Hawksworth, D.L., et al., *Dictionary of the Fungi*. 8th ed. 1995, Cambridge: Cambridge University Press.
6. Larone, D.H., *Medically Important Fungi: A Guide To Identification*. 3rd ed. 1995, Washignton, DC: ASM Press.
7. Kurtzman, C.P. and J.W. Fell, *The Yeasts, A Taxonomic Study*. 4th ed. 2000, Amsterdam: Elsevier Science BV.
8. Haynes, K., *Virulence in Candida species.* TRENDS in Microbiology, 2001. **9**(12): p. 591-596.
9. Hazen, K.C., *New an Emerging Yeast Pathogens.* Clinical Microbiology Reviews, 1995. **8**(4): p. 462-478.
10. Ampel, N.M., *Emerging Disease Issues and Fungal Pathogens Associated with HIV Infection.* Emerging Infectious Diseases, 1996. **2**(2): p. 109-116.
11. Lodish, H., et al., *Molecular Cell Biology*. 4th ed. 2000, New York: W. H. Freeman & Co.
12. Brown, T.A., *Genomes*. 2nd ed. 2002, Oxford: BIOS Scientific Publishers Ltd.
13. Berg, J.M., J.L. Tymoczko, and L. Stryer, *Biochemistry*. 5th ed. 2002, New York: W. H. Freeman & Co.
14. Strachan, T. and A.P. Read, *Human Molecular Genetics*. 2nd ed. 1999, Oxford: BIOS Scientific Publishers Ltd.
15. Alberts, B., et al., *Molecular Biology of the Cell*. 4th ed. 2002, New York: Garland Science Publishing.
16. Sugimoto, N., et al., *Improved thermodynamic parameters and helix initiation factor to predict stability of DNA duplexes.* Nucleic Acids Research, 1996. **24**(22): p. 4501-4505.
17. Breslauer, K.J., et al., *Predicting DNA duplex stability from the base sequence.* Proc. Natl. Acad. Sci., 1986. **83**: p. 3746-3750.
18. Owczarzy, R., et al., *Predicting Sequence-Dependent Melting Stability of Short Duplex DNA Oligomers.* Biopolymers, 1998. **44**(3): p. 217-239.
19. SantaLucia, J., Jr., *A unified view of polymer, dumbbell, and oligonucleotide DNA nearest-neighbor thermodynamics.* Proc. Natl. Acad. Sci., 1998. **95**: p. 1460-1465.
20. Burkard, M.E., D.H. Turner, and I. Tinoco, Jr., *The Interactions That Shape RNA Structure*, in *The RNA World*, R.F. Gesteland, T.R. Cech, and J.F. Atkins, Editors. 1999, Cold Spring Harbor Laboratory Press: Cold Spring Harbor, New York.
21. Van de Peer, Y., et al., *Database on the structure of small ribosomal subunit RNA.* Nucleic Acids Research, 1997. **25**(1): p. 111-116.
22. Darnell, J., H. Lodish, and D. Baltimore, *Molecular Cell Biology*. 2nd ed. 1990, New York: Scientific American Books.

23. Ban, N., et al., *The Complete Atomic Structure of the Large Ribosomal Subunit at 2.4 Å Resolution.* Science, 2000. **289**(5481): p. 905-920.
24. Venter, J.C., et al., *The Sequence of the human genome.* Science, 2001. **291**: p. 1304-1351.
25. IHGSC (Internacional Human Genome Sequencing Consortium), *Initial Sequencing and Analysis of the Human Genome.* Nature, 2001. **409**: p. 860-921.
26. AGI (The Arabidopsis Genome Initiative), *Analysis of the genome sequence of the flowering plant Arabidopsis thaliana.* Nature, 2000. **408**: p. 796-815.
27. CESC (The *C. elegans* Sequencing Consortium), *Genome sequence of the nematode C. elegans: a platform for investigating biology.* Science, 1998. **282**: p. 2012-2018.
28. Goffeau, A., et al., *Life with 6000 genes.* Science, 1996. **274**: p. 562-567.
29. Blattner, F.R., et al., *The complete genome sequence of Escherichia coli K-12.* Science, 1997. **277**: p. 1453-1462.
30. Nei, M. and S. Kumar, *Molecular Evolution and Phylogenetics*. 1st ed. 2000, New York: Oxford University Press, Inc.
31. Cai, J., I.N. Roberts, and M.D. Collins, *Phylogenetic relationships among members of the ascomycetous yeast genera Brettanomyces, Debaryomyces, Dekkera, and Kluyveromyces deduced by small-subunit rRNA gene sequences.* Int. J. Syst. Bacteriol., 1996. **46**(2): p. 542-549.
32. James, S.A., et al., *A phylogenetic analysis of the genus Saccharomyces based on 18S rRNA gene sequences: description of Saccharomyces kunashirensis sp. nov. and Saccharomyces martiniae sp. nov.* Int. J. Syst. Bacteriol., 1997. **47**(2): p. 453-460.
33. Kurtzman, C.P. and C.J. Robnett, *Identification and phylogeny of ascomycetous yeasts from analysis of nuclear large subunit (26S) ribosomal DNA partial sequences.* Antonie van Leeuwenhoek, 1998. **73**: p. 331-371.
34. Oda, Y., et al., *A phylogenetic analysis of Saccharomyces species by the sequence of 18S-28S rRNA spacer regions.* Yeast, 1997. **13**(13): p. 1243-1250.
35. James, S.A., M.D. Collins, and I.N. Roberts, *Use of an rRNA internal transcribed spacer region to distinguish phylogenetically closely related species of the genera Zygosaccharomyces and Torulaspora.* Int. J. Syst. Bacteriol., 1996. **46**(1): p. 189-194.
36. Kurtzman, C.P. and C.J. Robnett, *Phylogenetic relationships among yeasts of the 'Saccharomyces complex' determined from multigene sequence analyses.* FEMS Yeast Research, 2003. **3**(4): p. 417-432.
37. Van de Peer, Y., et al., *An Updated and Comprehensive rRNA Phylogeny of (Crown) Eukaryotes Based on Rate-Calibrated Evolutionary Distances.* J. Mol. Evol., 2000. **51**: p. 565-576.
38. Everett, K.D.E., R.M. Bush, and A.A. Andersen, *Emended description of the order Chlamydiales, proposal of Parachlamydiaceae fam. nov. and Simkaniaceae fam. nov., each containing one monotypic genus, revised taxonomy of the family Chlamydiaceae, including a new genus and five species, and standards for the identification of organisms.* Int. J. Syst. Bacteriol., 1999. **49**: p. 415-440.
39. Van de Peer, Y., S. Chapelle, and R. De Wachter, *A quantitative map of nucleotide substitution rates in bacterial ribosomal subunit RNA.* Nucleic Acids Research, 1996. **24**: p. 3381-3391.

40. Ali, A.B., et al., *Construction of a variability map for eukaryotic large subunit ribosomal RNA.* Nucleic Acids Research, 1999. **27**(14): p. 2825-2831.
41. Kellis, M., et al., *Sequencing and comparison of yeast species to identify genes and regulatory elements.* Nature, 2003. **423**: p. 241-254.
42. Cheung, V.G., et al., *Making and reading microarrays.* Nature Genetics, 1999. **21**: p. 15-19.
43. Holloway, A.J., et al., *Options available - from start to finish - for obtaining data from DNA microarrays II.* Nature Genetics, 2002. **32**: p. 481-489.
44. Southern, E., K. Mir, and M. Shchepinov, *Molecular interactions on microarrays.* Nature Genetics, 1999. **21**: p. 5-9.
45. Heller, M.J., *DNA Microarray Technology: Devices, Systems, and Applications.* Annu. Rev. Biomed. Eng., 2002. **4**: p. 129-153.
46. *IMAGE Consortium [http://www.image.llnl.gov/image/html/idistributors.shtml].*
47. Choudhuri, S., *Microarrays in Biology and Medicine.* J. Biochem. Molecular Toxicology, 2004. **18**(4): p. 171-179.
48. Bednar, M., *DNA microarray technology and application.* Med. Sci. Monit., 2000. **6**(4): p. 796-800.
49. Lipshutz, R.J., et al., *High density synthetic oligonucleotide arrays.* Nature Genetics, 1999. **21**: p. 20-24.
50. Murphy, D., *Gene expression studies using microarrays: principles, problems, and prospects.* Adv. Physiol. Educ., 2002. **26**(4): p. 256-270.
51. Barrett, J.C. and E.S. Kawasaki, *Microarrays: The use of oligonucleotides and cDNA for the analysis of gene expression.* Drug Discovery Today, 2003. **8**(3): p. 134-141.
52. Schena, M. and et al., *Parallel human genome analysis: microarray-based expression monitoring of 1000 genes.* Proc. Natl. Acad. Sci., 1996. **93**: p. 10614-10619.
53. Brown, P.O. and D. Botstein, *Exploring the new world of the genome with DNA microarrays.* Nature Genetics, 1999. **21**: p. 33-37.
54. Lashkari, D.A. and et al., *Yeast microarrays for genome wide parallel genetic and gene expression analysis.* Proc. Natl. Acad. Sci., 1997. **94**: p. 13057-13062.
55. DeRisi, J.L., V.R. Iyer, and P.O. Brown, *Exploring the metabolic and genetic control of gene expression on a genomic scale.* Science, 1997. **278**: p. 680-686.
56. Cho, R.J. and et al., *A genome-wide transcriptional analysis of the mitotic cell cycle.* Molecular Cell, 1998. **2**: p. 65-73.
57. Cole, K.A., D.B. Krizman, and M.R. Emmert-Buck, *The genetics of cancer - a 3D model.* Nature Genetics, 1999. **21**: p. 38-41.
58. Chung, C.H., P.S. Bernard, and C.M. Perou, *Molecular portraits and the family tree of cancer.* Nature Genetics, 2002. **32**: p. 533-540.
59. Hofman, P., *DNA microarrays: a practical approach from a pathologist's standpoint.* Nephron Physiol., 2005. **99**: p. 85-89.
60. Thomas, D.M., et al., *Molecular medicine: a clinician's primer on microarrays.* Internal Medicine Journal, 2004. **34**: p. 565-569.
61. Debouck, C. and P.N. Goodfellow, *DNA microarrays in drug discovery and development.* Nature Genetics, 1999. **21**: p. 48-50.
62. Gerhold, D.L., R.V. Jensen, and S.R. Gullans, *Better therapeutics through microarrays.* Nature Genetics, 2002. **32**: p. 547-552.

63. Petricoin, E.F., III and et al., *Medical applications of microarray technologies: a regulatory science perspective.* Nature Genetics, 2002. **32**: p. 474-479.
64. Weeraratna, A.T., et al., *Gene expression profiling: from microarrays to medicine.* J. Clinical Immunology, 2004. **24**(3): p. 213-224.
65. McCarthy, J.J. and R. Hilfiker, *The use of single-nucleotide polymorphism maps in pharmacogenomics.* Nature Biotech., 2000. **18**: p. 505-508.
66. Kellam, P., *Host-pathogen studies in the post-genomic era.* Genome Biology, 2000. **1**(2): p. 1009.1-1009.4.
67. Kellam, P., *Post-genomic virology: the impact of bioinformatics, microarrays and proteomics on investigating host and pathogen interactions.* Rev. Med. Virol, 2001. **11**: p. 313-329.
68. Bryant, P.A., et al., *Chips with everything: DNA microarrays in infectious diseases.* The Lancet - Infectious Diseases, 2004. **4**: p. 100-111.
69. Kato-Maeda, M., Q. Gao, and P.M. Small, *Microarray analysis of pathogens and their interaction with hosts.* Cellular Microbiology, 2001. **3**(11): p. 713-719.
70. Fradin, C., et al., *Stage-specific gene expression of Candida albicans in human blood.* Molecular Microbiology, 2003. **47**(6): p. 1523-1543.
71. Talaat, A.M., et al., *The temporal expression profile of Mycobacterium tuberculosis infection in mice.* Proc. Natl. Acad. Sci., 2004. **101**(13): p. 4602-4607.
72. Chambers, J., et al., *DNA microarrays of the complex human cytomegalovirus genome: profiling kinetic class with drug sensitivity of viral gene expression.* Journal of Virology, 1999. **73**(7): p. 5757-5766.
73. Betts, J.C., et al., *Signature gene expression profiles discriminate between Isoniazid- thiolactomycin-, and triclosan- treated Mycobacterium tuberculosis.* Antimicrobial Agents and Chemotherapy, 2003. **47**(9): p. 2903-2913.
74. Wilson, M., et al., *Exploring drug-induced alterations in gene expression in Mycobacterium tuberculosis by microarray hybridization.* Proc. Natl. Acad. Sci., 1999. **96**(22): p. 12833-12838.
75. De Backer, M.D., et al., *Genomic profiling of the response of Candida albicans to itraconazole treatment using a DNA microarray.* Antimicrobial Agents and Chemotherapy, 2001. **45**: p. 1660-1670.
76. Bammert, G.F. and J.M. Fostel, *Genome-wide expression patterns in Saccharomyces cerevisiae: comparison of drug treatments and genetic alterations affecting biosynthesis of ergosterol.* Antimicrobial Agents and Chemotherapy, 2000. **44**(5): p. 1255-1265.
77. Ng, W., et al., *Transcriptional regulation and signature patterns revealed by microarray analyses of Streptococcus pneumoniae R6 challenged with sublethal concentrations of translation inhibitors.* Journal of Bacteriology, 2003. **185**(1): p. 359-370.
78. Manger, I.D. and D.A. Relman, *How the host 'sees' pathogens: global gene expression responses to infection.* Curr. Opin. Immunol, 2000. **12**: p. 215-218.
79. McGuire, K. and E.J. Glass, *The expanding role of microarrays in the investigation of macrophage responses to pathogens.* Veterinary Immunology and Immunopathology, 2005. **105**: p. 259-275.
80. Eckmann, L., et al., *Analysis by high density cDNA arrays of altered gene expression in human intestinal epithelial cells in response to infection with the invasive enteric bacteria Salmonella.* The Journal of Biological Chemistry, 2000. **275**(19): p. 14084-14094.

81. Leong, W.F., et al., *Microarray and real-time RT-PCR analysis of differential human gene expression patterns induced by severe acute respiratory syndrome (SARS) coronavirus infection of Vero cells.* Microbes and Infection, 2005. **7**: p. 248-259.
82. Zhu, H., et al., *Cellular gene expression altered by human cytomegalovirus: global monitoring with oligonucleotide arrays.* Proc. Natl. Acad. Sci., 1998. **95**: p. 14470-14475.
83. Lovmar, L., et al., *Microarrays for genotyping human group A rotavirus by multiplex capture and type-specific primer extension.* J. Clinical Microbiology, 2003. **41**(11): p. 5153-5158.
84. Chizhikov, V., et al., *Detection and genotyping of human group A rotaviruses by oligonucleotide microarray hybridization.* J. Clinical Microbiology, 2002. **40**(7): p. 2398-2407.
85. Wang, D., et al., *Microarray-based detection and genotyping of viral pathogens.* Proc. Natl. Acad. Sci., 2002. **99**(24): p. 15687-15692.
86. Sengupta, S., et al., *Molecular detection and identification of influenza viruses by oligonucleotide microarray hybridization.* J. Clinical Microbiology, 2003. **41**(10): p. 4542-4550.
87. Chizhikov, V., et al., *Microarray analysis of microbial virulence factors.* Appl. Environ. Microbiol., 2001. **67**(7): p. 3258-3263.
88. Kakinuma, K., M. Fukushima, and R. Kawaguchi, *Detection and identification of Escherichia coli, Shigella, and Salmonella by microarrays using the gyrB gene.* Biotechnol. Bioeng., 2003. **83**(6): p. 721-728.
89. Wang, R.F., et al., *Design and evaluation of oligonucleotide-microarray method for the detection of human intestinal bacteria in fecal samples.* FEMS Microbiol. Lett., 2002. **213**: p. 175-182.
90. Volokhov, D., et al., *Identification of Listeria species by microarray-based assay.* J. Clinical Microbiology, 2002. **40**(12): p. 4720-4728.
91. Volokhov, D., et al., *Microarray-based identification of thermophilic Campylobacter jejuni, C. coli, C. lari, and C. upsaliensis.* J. Clinical Microbiology, 2003. **41**(9): p. 4071-4080.
92. Anthony, R.M., T.J. Brown, and G.L. French, *Rapid diagnosis of bacteremia by universal amplification of 23S ribosomal DNA followed by hybridization to an oligonucleotide array.* J. Clinical Microbiology, 2000. **38**(2): p. 781-788.
93. Rudi, K., et al., *Development and evaluation of a 16S ribosomal DNA array-based approach for describing complex microbial communities in ready-to-eat vegetable salads packed in a modified atmosphere.* Appl. Environ. Microbiol., 2002. **68**(3): p. 1146-1156.
94. Wilson, K.H., et al., *High-density microarray of small-subunit ribosomal DNA probes.* Appl. Environ. Microbiol., 2002. **68**(5): p. 2535-2541.
95. Bodrossy, L., et al., *Development and validation of a diagnostic microbial microarray for methanotrophs.* Environ. Microbiol., 2003. **5**(7): p. 566-582.
96. Loy, A., et al., *Oligonucleotide microarray for 16S rRNA gene-based detection of all recognized lineages of sulfate-reducing prokaryotes in the environment.* Appl. Environ. Microbiol., 2002. **68**(10): p. 5064-5081.
97. El Fantroussi, S., et al., *Direct profiling of environmental microbial populations by thermal dissociation analysis of native rRNAs hybridized to oligonucleotide microarrays.* Appl. Environ. Microbiol., 2003. **69**(4): p. 2377-2382.

98. Denef, V.J., et al., *Validation of a more sensitive method for using spotted oligonucleotide DNA microarrays for functional genomics studies on bacterial communities.* Environ. Microbiol., 2003. **5**(10): p. 933-943.
99. Tiquia, S.M., et al., *Evaluation of 50-mer oligonucleotide arrays for detecting microbial populations in environmental samples.* BioTechniques, 2004. **36**(4): p. 664-675.
100. Wuyts, J., et al., *The European large subunit ribosomal RNA database.* Nucleic Acids Research, 2001. **29**(1): p. 175-177.
101. Wuyts, J., et al., *The European database on small subunit ribosomal RNA.* Nucleic Acids Research, 2002. **30**(1): p. 183-185.
102. Markoulatos, P., N. Siafakas, and M. Moncany, *Multiplex Polymerase Chain Reaction: a practical approach.* J. Clin. Lab. Analysis, 2002. **16**: p. 47-51.
103. Quackenbush, J., *Computational analysis of microarray data.* Nature Reviews Genetics, 2001. **2**: p. 418-427.
104. Eschrich, S. and T.J. Yeatman, *DNA microarrays and data analysis: an overview.* Surgery, 2004. **136**(3): p. 500-503.
105. Miller, L.D., et al., *Optimal gene expression analysis by microarrays.* Cancer Cell, 2002. **2**: p. 353-361.
106. Brazma, A., et al., *One-stop shop for microarray data.* Nature, 2000. **403**: p. 699-700.
107. Stoeckert, C.J., Jr., H.C. Causton, and C.A. Ball, *Microarray databases: standards and ontologies.* Nature Genetics, 2002. **32**: p. 469-473.
108. Brazma, A. and et al., *Minimum information about a microarray experiment (MIAME) - toward standards for microarray data.* Nature Genetics, 2001. **29**: p. 365-371.
109. Spellman, P.T. and et al., *Design and implementation of microarray gene expression markup language (MAGE-ML).* Genome Biology, 2002. **3**(9): p. research0046.1-0046.9.
110. Parkinson, H. and et al., *ArrayExpress - a public repository for microarray gene expression data at the EBI.* Nucleic Acids Research, 2005. **33**: p. 553-555.
111. Edgar, R., M. Domrachev, and A.E. Lash, *Gene Expression Omnibus: NCBI gene expression and hybridization array data repository.* Nucleic Acids Research, 2002. **30**(1): p. 207-210.
112. Ikeo, K., et al., *CIBEX: Center for Information Biology gene EXpression database.* C. R. Biologies, 2003. **326**: p. 1079–1082.
113. Ball, C.A. and et al., *Submission of microarray data to public repositories.* PLoS Biology, 2004. **2**(9): p. 1276-1277.
114. Baxevanis, A.D. and B.F.F. Ouellette, *Bioinformatics: a practical guide to the analysis of genes and proteins*. 2nd ed. 2001, New York: Wiley-Interscience.
115. Mount, D.W., *Bioinformatics: Sequence and Genome Analysis*. 1st ed, ed. J. Cuddihy. 2001, Cold Spring Harbor, New York: Cold Spring Harbor Laboratory Press.
116. Wheeler, D.L. and et al., *Database resources of the National Center for Biotechnology Information.* Nucleic Acids Research, 2005. **33**(Database issue): p. D39-D45.
117. Altschul, S.F., et al., *Basic local alignment search tool.* J. Mol. Biol., 1990. **215**: p. 403-410.

118. Altschul, S.F., et al., *Gapped BLAST and PSI-BLAST: a new generation of protein database search programs.* Nucleic Acids Research, 1997. **25**(17): p. 3389-3402.
119. Madden, T.L., *The BLAST sequence analysis tool*, in *The NCBI Handbook [http://www.ncbi.nlm.nih.gov]*, J. McEntyre, Editor. 2003, National Library of Medicine (US), National Center for Biotechnology Information, Bethesda, MD.
120. McGinnis, S. and T.L. Madden, *BLAST: at the core of a powerful and diverse set of sequence analysis tools.* Nucleic Acids Research, 2004. **32**(Web server issue): p. W20-W25.
121. Thompson, J.D., D.G. Higgins, and T.J. Gibson, *CLUSTAL W: improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice.* Nucleic Acids Research, 1994. **22**(22): p. 4673-4680.
122. Chenna, R., et al., *Multiple sequence alignment with the Clustal series of programs.* Nucleic Acids Research, 2003. **31**(13): p. 3497-3500.
123. Thompson, J.D., et al., *The CLUSTAL_X windows interface: flexible strategies for multiple sequence alignment aided by quality analysis tools.* Nucleic Acids Research, 1997. **25**(24): p. 4876-4882.
124. Dwyer, R.A., *Genomic PERL: from bioinformatics basics to working code*, ed. L. Cowles. 2003, Cambridge: Cambridge University Press.
125. Stajich, J.E. and et al., *The Bioperl toolkit: Perl modules for the life sciences.* Genome Research, 2002. **12**: p. 1611-1618.
126. Chee, M. and et al., *Accessing Genetic Information with High-Density DNA Arrays.* Science, 1996. **274**(5287): p. 610-614.
127. Kane, M.D., et al., *Assessment of the sensitivity and specificity of oligonucleotide (50mer) microarrays.* Nucleic Acids Research, 2000. **28**(22): p. 4552-4557.
128. Fotin, A.V., et al., *Parallel thermodynamic analysis of duplexes on oligodeoxyribonucleotide microchips.* Nucleic Acids Research, 1998. **26**(6): p. 1515-1521.
129. Urakawa, H., et al., *Optimization of single-base-pair mismatch discrimination in oligonucleotide microarrays.* Appl. Environ. Microbiol., 2003. **69**(5): p. 2848-2856.
130. Urakawa, H., et al., *Single-base-pair discrimination of the terminal mismatches by using oligonucleotide microarrays and neural network analyses.* Appl. Environ. Microbiol., 2002. **68**(1): p. 235-244.
131. Peplies, J., F.O. Glockner, and R. Amann, *Optimization strategies for DNA microarray-based detection of bacteria with 16S rRNA-targeting oligonucleotide probes.* Appl. Environ. Microbiol., 2003. **69**(3): p. 1397-1407.
132. Mannarelli, B.M. and C.P. Kurtzman, *Rapid identification of C. albicans and other human pathogenic yeasts by using short oligonucleotides in a PCR.* J. Clinical Microbiology, 1998. **36**(6): p. 1634-1641.
133. Nelson, B.P., et al., *Label-free detection of 16S ribosomal RNA hybridization on reusable DNA arrays using surface plasmon resonance imaging.* Environ. Microbiol., 2002. **4**(11): p. 735-743.
134. Chandler, D.P., et al., *Sequence versus structure for the direct detection of 16S rRNA on planar oligonucleotide microarrays.* Appl. Environ. Microbiol., 2003. **69**(5): p. 2950-2958.